MAIA
ERASMUS MUNDUS
JOINT MASTER IN MEDICAL IMAGING AND APPLICATIONS

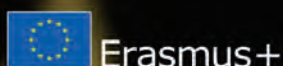# Joint Master in Medical Imaging and Applications

Master Thesis Proceedings

Promotion 2020-22

www.maiamaster.org

MAIA

Editorial

Computer aided applications for early detection and diagnosis, histopathological image analysis, treatment planning and monitoring, as well as robotised and guided surgery will positively impact health care during the new few years. The scientific community needs of prepared entrepreneurships with a proper ground to tackle these topics. The Joint Master Degree in Medical Imaging and Applications (MAIA) was born with the aim to fill this gap, offering highly skilled professionals with a depth knowledge on computer science, artificial intelligence, computer vision, medical robotics, and transversal topics.

The MAIA master is a two-years joint master degree (120 ECTS) between the Université de Bourgogne (uB, France), the Università degli studi di Cassino e del Lazio Meridionale (UNICLAM, Italy), and the Universitat de Girona (UdG, Spain), being the latter the coordinating institution. The program is supported by associate partners, that help in the sustainability of the program, not necessarily in economical terms, but in contributing in the design of the master, offering master thesis or internships, and expanding the visibility of the master. Moreover, the program is recognised by the European Commission for its academic excellence and is included in the list of Erasmus Mundus Joint Master Degrees under the Erasmus+ programme.

This document shows the outcome of the master tesis research developed by the MAIA students during the last semester, where they put their learnt knowledge in practice for solving different problems related with medical imaging. This include fully automatic anatomical structures segmentation, abnormality detection algorithms in different imaging modalities, biomechanical modelling, development of applications to be clinically usable, or practical components for integration into clinical workflows. We sincerely think that this document aims at further enhancing the dissemination of information about the quality of the master and may be of interest to the scientific community and foster networking opportunities amongst MAIA partners.

We finally want to thank and congratulate all the students for their effort done during this last semester of the Joint Master Degree in Medical Imaging and Applications.
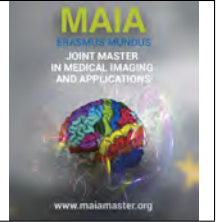
MAIA Master Academic and Administrative Board

Contents

NesT UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial Infarction Evaluation

Mahmoud Abdelhamed, Fabrice Meriaudeau

[a]mahmoud.k.nasr@gmail.com
[b]fabrice.meriaudeau@u-bourgogne.fr

Abstract

Myocardial Infarction (MI), commonly known as heart attack, is the irreversible death of the Myocardium's tissue due to the lack of oxygen for an extended period of time. *LGE-MRI* scans are considered the defacto in the diagnosis and prognosis of *MI*. Still, they require manually segmenting the Myocardium and the infarcted tissue, which is a complex and time-consuming task. Hence, an automatic segmentation method of the Myocardium tissue is highly desirable. CNNs (Convolutional Neural Network) are used extensively for solving this problem; over the *EMIDEC* (automatic Evaluation of Myocardial Infarction from Delayed Enhancement Cardiac MRI) challenge in *MICCAI* 2000, CNNs were used. Still, they required complex architectures to achieve state-of-the-art results. This paper presents a novel architecture based on *Self-Attention Transformer*. Vision transformers following the original vision transformer *ViT* have proven their capabilities in achieving state-of-the-art results on multiple benchmarks in different vision tasks, including medical imaging tasks beating many CNN architectures. Still, they require huge amount of data for either pretraining or training to achieve good performance which is a big restriction in applying them in the medical imaging domain despite their performance. This thesis introduces a novel segmentation architecture based on the *NesT* architecture for the encoder network and inspired by it we introduce a novel decoder based on the same architecture. *NesT* achieved state-of-the-art results on *ImageNet* and *CIFAR* classification tasks with minimal training compared to other transformer networks. In this paper we introduce *NesT-UNet* which produced results comparable to the state-of-the-art on the *EMIDEC* dataset using a simple training process including simple data augmentation and a pretraining method to improve the network's performance.

Keywords: Vision Transformers, Deep Learning, LGE-MRI, Cardiac Infarction Segmentation

1. Introduction

According to the World health organization *(WHO)*, Cardiovascular Diseases *(CVDs)* are the leading cause of death worldwide. An estimated 17.9 million people died from CVDs in 2019, representing 32% of all global deaths; of these deaths, 85% were due to heart attack and stroke[1].

Heart attack or Myocardial infarction *MI*[2], refers to the tissue death (infarction) of the heart muscle (myocardium). The death of the tissue happens due to *is-chemia*, which is the lack of oxygen supply to the heart tissues. Coronary arteries are responsible for supplying the heart with oxygenated blood flow. Still, when plaque starts to build up in an artery, narrowing it down, the blood flow becomes slower or, in some cases, completely blocked.

If the blockage in the artery is not treated immediately, the oxygen-deprived tissues around the artery start to die. The treatment, in this case, is revascularization, which is a procedure that tries to restore the blood flow in the blocked arteries. In some cases, the blood flow will not be restored completely in some regions. This phenomenon is called *No-Reflow*.

After the treatment, it is crucial to evaluate the state of the heart to check if the infarcted regions have re-

---

[1]https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

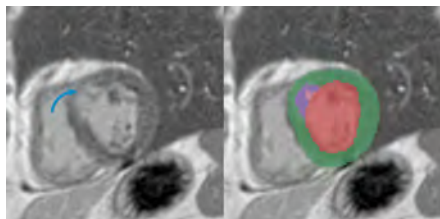[2]https://simple.wikipedia.org/wiki/Myocardial_infarction

Figure 1: LGE-MRI scan from the EMIDEC dataset where the colors overlaying the ground truth mask are representing the segmentation classes, Red for Left ventricle, Green for Myocardium, Purple for Infarction, and Pink for No-Reflow. The figure shows the difference in contrast between the healthy tissue and the Infarcted tissues

covered their functionalities after the revascularization. Cardiac MRI can be used to assess the state of the heart in a non-invasive and accurate manner, especially the late gadolinium enhancement *(LGE)* MRI.

LGE-MRI is considered the standard modality for detecting and evaluating MI. After injecting the patient with a gadolinium-based contrast agent for approximately 10 minutes, the scan is performed. The wash-in and wash-out of the contrast agent differentiate the healthy and infarcted tissues as presented in figure 1. In addition, the geometry of the Myocardium can be used to conclude the functionality of the damaged muscle, which guides further future treatment.

In the typical workflow, the scans are segmented by a clinician manually, which is an exhausting, time-consuming process. In addition, it suffers from inter- and intra-observer variability. These problems can be addressed by automatic cardiac segmentation of the healthy and damaged tissues.

That being said, automatic cardiac segmentation has many challenges, such as size variation of the segmented region due to the expansion and contraction of the heart during the different cardiac cycle phases and different shapes over heart regions such as basal, middle and apex slices. In addition, motion artifacts, low contrast between infarctions, healthy tissues, and the blood pool in the left ventricle *(LV)*, also the class imbalance between the infarctions and healthy tissues which make developing an automatic cardiac segmentation system a challenging problem.

In this paper, We explore a novel architecture for cardiac tissue segmentation. This document is organized as following, In section 2, we explore the literature covering this topic, establishing the state-of-the-art in the process. Also we explore the literature of transformers in general and in segmentation specifically. In section 3, we present the different approaches and experiments done during this work the we end the section with the proposed pipeline for segmentation. In section 5, we present the results following the experiments discussed in 3 with statistical information about the results. A final discussion summarizing the achieved results and reporting the final overall results in section 7.

1.1. Cardiac Structures Segmentation

In clinical cardiology, it is essential to measure and evaluate the state of the heart. Several metrics are crucial for the clinician to be able to diagnose and treat patients, such as Ejection Fraction *(EF)*, Stroke Volume, and myocardium thickness, and Cardiac MRI has become the standard for analysis. All the metrics depend on the accurate segmentation of the anatomical structures of the heart, which is a time-consuming activity.

Many datasets have been released with their respected challenges to find more robust and automated systems for solving the segmentation problem, such as The Sunnybrook Cardiac MR Left Ventricle Segmentation challenge - MICCAI 2009[3], The LV Segmentation Dataset and Challenge, MICCAI-STACOM 2011, The Right Ventricle Segmentation Dataset - MICCAI 2012[4], The 2015 Kaggle Second Annual Data Science Bowl, and Automated Cardiac Diagnosis Challenge *(ACDC)* - MICCAI 2017[5]. Some of these datasets predate the deep learning era, so all the applied techniques are based on the classical methods, but the number can show the importance of the problem and its difficulty.

1.2. EMIDEC

The *EMIDEC* (automatic Evaluation of Myocardial Infarction from Delayed Enhancement Cardiac MRI) challenge was organized during the *MICCAI* 2020 conference. The objectives of this challenge are to segment and classify cases with MI. The overall dataset consists of 150 exams, with 100 cases for training and the other 50 for testing. The training dataset contains 67 pathological cases, meaning the patient suffers from MI and 33 normal cases, while the testing dataset has 33 pathological cases and 17 normal cases. The dataset provides LGE-MRI exams composed of a series of short-axis slices and the associated clinical information. For each case, The segmentation mask shows the LV, Myocardium and if it is a pathological case, the segmentation mask provides both the MI and No Reflow tissue. Here we are focusing on the segmentation part of the challenge. The goal of the segmentation contest is to present the best automatic segmentation method for the Myocardium, the MI, and No Reflow tissue if they exist. Some samples of the dataset are shown in figure 2, The dataset is highly complex, for example in terms of size, we observe high variability aparent in the first row of figure 2 between the Apex slice, Middle Slices and Basal slice also the Myocardium thickness between them. In addition to the class imbalance between the presented classes because not all slices contain No-Reflow or Infarction and not a infected slice present the same malignant tissue size.

---

[3]http://www.cardiacatlas.org/studies/sunnybrook-cardiac-data/

[4]http://www.cardiacatlas.org/challenges/lv-segmentation-challenge/

[5]https://www.creatis.insa-lyon.fr/Challenge/acdc/

NesT UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial Infarction Evaluation
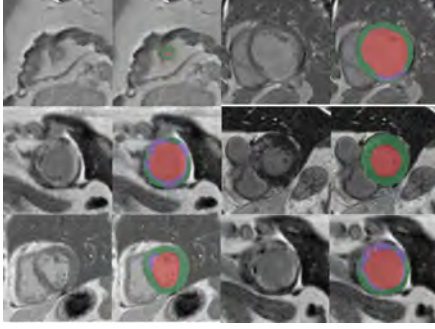
3

Figure 2: Samples from the EMIDEC dataset where the colors overlaying the ground truth mask are representing the segmentation classes, Red for Left ventricle, Green for Myocardium, Purple for Infarction, and Pink for No-Reflow. The figure presents the huge variability in size between the Apex, Middle and Basal Slices. Also it presents the variability of the infarction tissue which has irregular shape and irregular number of pixels within each slice

2. Literature Review

2.1. Convolution-Based algorithms

The problem of cardiac segmentation has evolved through multiple stages; the first was the use of Non-deep learning methods to segment different anatomical parts of the heart, such as the Right Ventricle *(RV)* and Left Ventricle *(LV)*. With the rise of deep learning,*CNNs* has achieved great success for Cardiac MRI segmentation.

For the past few years, CNNs have been used extensively for Cardiac MRI segmentation; in this section, we will discuss some of the methodologies used with a focus on the *EMIDEC* challenge since it is a recent challenge; it presents the mainstream research directions in cardiac segmentation and diagnosis. Lalande et al. (2022) presented the challenge results and provided an overview of the methodologies used for this challenge and cardiac segmentation in general.

CNN-based UNets had become the defacto architecture in solving the segmentation problem in medical imaging, and cardiac segmentation is no different. In the *EMIDEC* challenge, Challengers used various configurations of UNets. Zhang (2020), the winner of the challenge used a cascade of 2D-3D UNet, shown in figure 3, inspired by *nnUNet* (Isensee et al., 2021). The cascade aims to utilize the best properties in 2D and 3D. 2D UNet focuses only on the intra-slice features, hence 3D UNet refines the 2D segmentation; this concept avoids the intra-slice heterogeneity and considers the volumetric information for more refined segmentation.

Another method used was multi-stage segmentation, where the first stage is to segment the Myocardium and the second is to segment the infarction and no-reflow tissues. Camarasa et al. (2020) followed this method adding uncertainty to the process by passing the Myocardium segmentation to a probabilistic auto-encoder
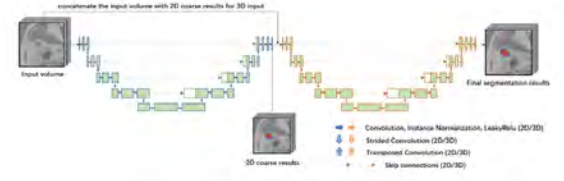
Figure 3: Zhang (2020) Schematic showing the cascaded network where the left part is the 2D UNet while the right part is the 3D UNet
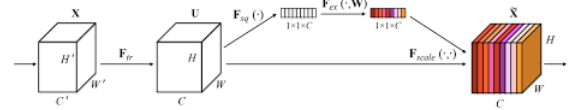
Figure 4: Hu et al. (2018), Schematics of Squeeze-and-Excitation module

using *Monte-Carlo* dropout, the generated map is fed to the second model for scar segmentation.

In addition to the more vanilla encoding blocks used with UNets such as *ResNet*, *ResNeXt*, and *Inception* modules, some challengers tried to add attention to the UNet encoders. Yang and Wang (2020), and Girum et al. (2020) applied *Squeeze-and-excitation (SE)* (Hu et al., 2018) for better modeling capabilities. Girum et al. (2020) also used the *Selective Kernel (SK)* block in the decoder to adaptively adjust the receptive field size to enable automatic kernel size selection.

SE was one of the first adaptors of attention mechanism in CNNs, SE blocks managed to improve channel inter-dependencies at almost no computational cost by adaptively adjusting the weight of each feature map in the aggregation of the final map. The input for the *SE* block is a feature map of size $C \times H \times W$, where C is the number of channels, (H, W) are the spatial dimensions of the feature map. The feature map's spatial dimensions are *Squeezed*, usually, by average pooling into a vector of shape $C \times 1 \times 1$. Fully connected layers project the squeezed map into weights to aggregate the original feature maps into the final output map, focusing on the important feature maps by *Exciting* them with large weight values. Figure 4 illustrates the schematics of this block. The *SE* block with its attention mechanism improved many computer vision algorithms but it can only be considered as an addition to the convolution blocks because the modeling and the feature extraction process still depends on the convolution layers.

Challenge discussion. The challenge results are shown in table 1. The overall segmentation of the Myocardium is good, where the best performance was a 0.879 Dice score and 0.712 Dice score for the Infarction, but on the other hand, the No-Reflow tissue proved to be difficult, with the best Dice score at 0.785. It is also worth noting that the Dice score for the No-Reflow class is equal to one for the correct classification of its absence from the

NeST UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial
Infarction Evaluation

4

| Challenger	Myocardium			Infarction			No-Reflow				
	Dice	Vol Diff. (cm^3)	Hausdorff (mm)	Dice	Vol Diff. (cm^3)	Pct. Diff. (%)	Dice	Vol Diff. (cm^3)	Pct. Diff (%)	Acc. (case (%))	Acc. (slice (%))
Zhang	0.879±0.027	9.26±9.08	13.01±8.81	0.712±0.268	3.12±5.15	2.38±0.031	0.785±0.393	0.63±2.27	0.38±0.012	84.00	94.97
Feng et al.	0.836±0.124	15.19±16.41	33.77±111.63	0.547±0.340	3.97±8.36	2.89±0.045	0.722±0.432	0.88±3.41	0.53±0.017	80.00	90.78
Yang et al.	0.855±0.027	16.54±10.27	13.23±6.80	0.628±0.315	5.34±7.88	4.37±0.062	0.610±0.463	1.85±3.32	1.69±0.033	76.00	81.56
Huellebrand et al.	0.841±0.051	10.87±8.53	18.3±15.74	0.379±0.296	6.17±8.36	4.93±0.059	0.523±0.483	0.95±3.00	0.64±0.015	70.00	85.75
Camarasa et al.	0.757±0.111	17.11±15.45	25.44±21.71	0.308±0.280	4.87±8.49	3.64±0.047	0.605±0.485	0.87±3.27	0.52±0.016	74.00	84.36
Zhou et al.	0.825±0.057	13.29±11.34	83.42±158.97	0.378±0.309	6.10±9.45	4.71±0.06	0.520±0.487	0.88±3.38	0.54±0.017	64.00	86.87
Brahim et al.3	0.791±0.050	12.68±10.59	23.87±11.52	0.274±0.379	7.05±12.73	5.19±0.074	0.641±0.479	0.83±3.109	0.50±0.016	74.00	89.39
Girum et al.3	0.803±0.057	11.81±14.09	51.48±98.15	0.340±0.474	11.52±16.53	8.58±0.101	0.780±0.414	0.89±3.61	0.51±0.018	78.00	89.66

Table 1: Lalande et al. (2022), *EMIDEC* Challenge results. Pct. Diff.: Difference between the percentage of the infarcted Myocardium.
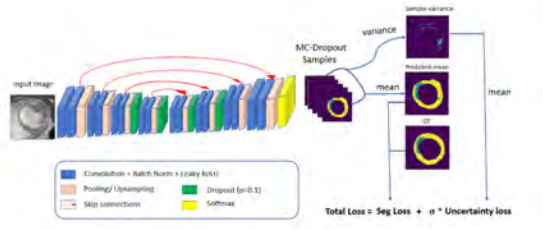
Figure 5: Arega et al. (2021), Schematics of the used method with the losses used
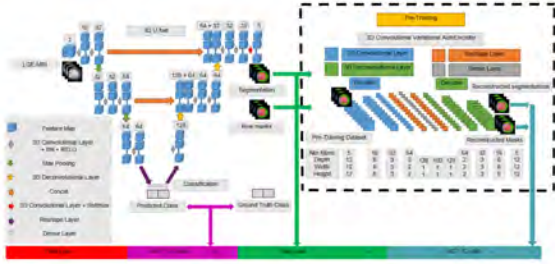
Figure 6: Brahim et al. (2022), Schematics of the ICPIU-Net architecture and training

case, which points out the difficulty of the task. From the results, we can deduce that more complex networks and pipelines are not guaranteed to produce better results; on the contrary, more adaptive and more tuned architectures are achieving better results following the *nnUNet* philosophy contradicting the ideas in the natural images for datasets such as *CityScape* or *COCO*.

Other researchers tried to enhance the segmentation results after the challenge. Arega et al. (2021) proposed to use *Monte-Carlo (MC)* dropout within the network to generate N samples, then the final prediction is the average of the samples, and the variance of the samples was used as an uncertainty loss as shown in figure 5.

Brahim et al. (2022) used a multi-stage strategy for segmentation, as shown in figure 6. The first stage is 3D UNet for initial segmentation with an extra classification head to recognize the pathological cases. The second stage is refining the shape of the Myocardium using a 3D convolutional variational auto-encoder to reconstruct the prediction mask. An ensemble of models is used to compute the final prediction mask. To our knowledge, the *ICPIU-Net* achieves the best results on the *EMIDEC* dataset with Myocardium Dice of 95.32, Infarction Dice of 78.3, and No-Reflow Dice of 77.83.

So far, all the architectures used depends on some fusion technique. Multi-stage networks are trying to overcome the lack of a single convolution 2D or 3D to predict the output well enough, so the prediction of each stage is refined or utilized by the next step to overcome its shortcomings, inspiring going in a different direction. With the rise of self-attention-based networks (transformers), it seems like an excellent candidate to solve the misgivings of convolution-based UNets.

2.2. Transformers

The first transformer emerged from the natural language processing *(NLP)* field in the sequence-to-sequence application such as machine translation. They emerged from the need to model the long-term dependencies between elements in the sequence, i.e., words in a sentence to generate the appropriate output sequence. Now transformers dominate the field of *NLP* and are applied in many applications within the *NLP* field and other fields.

Transformers were initially designed to solve the machine translation problem and were explored extensively in the literature Fan et al. (2021), Mehta et al. (2020), and So et al. (2019). Transformers were also adopted in other applications such as language modeling (Dai et al., 2019), (Rae et al., 2019), and named entity recognition Li et al., Yan et al. (2019).

Also, a significant effort was focused on pretraining models on a very large scale that can serve as a start point for many applications such as the *GPT* series, *GPT* (Radford et al., 2018), *GPT-2* (Radford et al., 2019), and lastly *GPT-3* (Brown et al., 2020). *GPT-3* is a 175 billion parameter model that is being used for many applications such as translation and question answering. also, it has been used to write new articles from scratch and generate code. *BERT* Devlin et al. (2018) is another example, enabling anyone to develop their question answering system achieving state-of-the-art results.

Transformers also was adopted in other fields; In audio applications, transformers have been used for speech recognition (Chen et al., 2021e), (Dong et al., 2018), Speech synthesis (Ihm et al., 2020), (Zheng et al., 2020) and many other applications. Lin et al. (2021) is a comprehensive review of transformer applications.

2.2.1. Self-Attention

Transformers layers depend on the *self-attention* mechanism. The attention is a function mapping a se-

Scaled Dot-Product Attention

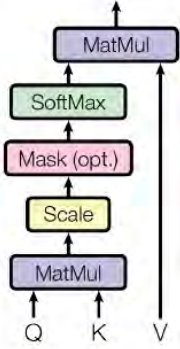Figure 7: Vaswani et al. (2017) Schematic showing the self-attention mechanism

quence of input vectors to a corresponding output sequence where each output vector is a weighted sum of all the input vectors. A compatibility function computes the weights depending on the relation between each vector with the other elements in the input sequence.

Vaswani et al. (2017), presents the attention function as a mapping between a query and a set of key-value pairs, as shown in figure 7. The *Scaled Dot-Product Attention* compatibility function computes a score between each query and each of the key-value pairs. The function calculates the dot product between the query vector and each of the key vectors, and then the resulting product will be normalized by the square root of the vector's size d to neutralize the effect of the vector's length on the dot product's value range. A *Softmax* function is applied to the scaled dot product vectors to get the final weights used to get the output vector.

Practically, The computations are done in a matrix format to speed up the calculations by stacking the queries into one matrix Q and the key-value pairs into matrices K and V; the final output matrix is:

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d}})V \quad (1)$$

2.2.2. Multi-Head Attention

Similar to *CNNs*, where each convolution layer applies multiple kernels to extract various features from the same input, each attention layer applies multiple *attention heads* to the same input where each attention head is using equation 1. Each attention head is applied to a projected part of the input vector. Each head attends to a subspace of the original feature vector extracting different information independently, increasing the layer's modeling capability. The outputs of all the heads are concatenated and then are linearly projected into the output embedding dimension. The formulation

is as follows:

$$MultiHead(Q, K, V) = Concat(H_1, \cdots, H_h)W^O \quad (2)$$

where, H is one of the head

$$H_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3)$$

A linear, fully connected layer can model each projection for batch implementation.

2.2.3. Position Embedding

The attention mechanism doesn't account for the elements' order in the sequence, meaning that all the permutations of the sequence will result in the same output vectors. Position embeddings solve this problem by using the absolute position of each vector in the sequence to compute an embedding that is added to the input vector accounting for the vector's position in the sequence.

Vaswani et al. (2017) used a fixed position embedding equation found empirically and showed that the results of using learnable position parameters are almost identical to the fixed embedding function. This is true for different *NLP* applications, but other works produce better results with learnable position encoding parameters.

2.3. Vision Transformers

Transformers and attention mechanism proved their modeling power in the *NLP* field. Inspired by their success, multiple works tried to adopt transformer architectures in computer vision. Dosovitskiy et al. (2020) were the first to try applying pure transformer architecture in computer vision with almost no modification in the original architecture. They based their design on the original transformer paper Vaswani et al. (2017). The only modification was transforming the image into a sequence because the nature of transformers requires the input to be a sequence of elements. An image is a 2D sequence of pixels, but using the self-attention module on the pixel level will be prohibitive in computation complexity and memory requirements. Hence, the image was split into 16×16 patches then each patch was embedded into a smaller dimension space using an *embedding layer*. A linear projection function, a fully connected layer, was used to perform the patch embedding where each patch was reduced into a D dimension vector and will stay constant across the transformer layers in the network.

Each transformer block is composed of multi-headed self-attention *(MSA)* followed by an multi-layer perceptron block *(MLP)*. The layer's architecture is shown in figure 8 with an overview of the model. The results from the *ViT* architecture triggered many researchers to incorporate transformers architectures in different computer vision tasks from the recognition tasks such as image classification and segmentation to multi-model

NesT UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial Infarction Evaluation
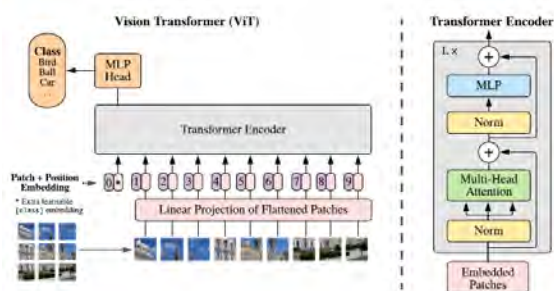
6

Figure 8: Dosovitskiy et al. (2020), Left section is an overview of the model, the right section is the architecture transformer encoder

problems such as visual-question answering, visual reasoning, Khan et al. (2021) prove a comprehensive survey about the different applications of *ViTs* in the computer vision domain.

2.4. Vision Transformers in Medical Imaging

Medical imaging started adopting transformers and applying them in different applications for diagnosis and prognosis. He et al. (2022) is a comprehensive review of many applications using transformer-based architectures in the medical imaging field.

2.4.1. Classification

In the classification problem, some applications applied *ViT* as is. Gheflati and Rivaz (2021) used ViT on a breast ultrasound dataset to classify normal, malignant, and benign breast tissues. They compared the ViT fine-tuned network vs. many CNN based models such as *ResNet*, *VGG*, and *Inception*, and they reported better performance from the *ViT* architecture on both accuracy and area under the curve *(AUC)*.

Gao et al. (2021a) compared *ViT* and *DenseNet* on COVID-19 CT-Scan diagnosis dataset. The dataset has both 2D and 3D scans, and it is worth noting that they extracted sub-volumes from the 3D scan, effectively fixing the sequence length for the *ViT* and solving the problem of variable 3D scan depth. There results showed that *ViT* performance is better than *DenseNet*.

other contributions tried to utilize the ideas of the vision transformer. Liu and Yin (2021) applied ViT-based architecture *VOLO* Yuan et al. (2021) for COVID-19 diagnosis from *X-Ray* images. *VOLO* implements a new kind of attention called *outlooker attention* achieving SOTA performance for COVID-19 diagnosis.

Other methods try to combine self-attention layers with other components, such as convolution layers which have a high inductive bias for images and have the potential to increase the data efficiency either for training or fine-tuning. Also, from a modeling standpoint, attention tries to model the relationship between elements of the sequence, while convolution focuses on the local features extracted from the neighborhood of

each pixel. The difference in the modeling techniques is encouraging to integrate both of them.

Barhoumi and Ghulam (2021), Used an ensemble of different CNN networks, trained using different paradigms to extract diverse and rich features, then the features are fed to a transformer encoder for classification. They showed that the results increase with the number of CNNs used and the quality of each; also, the modeling power of the transformer encoder captured the important information from each CNN's feature map.

Chen et al. (2021a), proposed *GasHis-Transformer*, to classify gastric histopathological images. They used a multi-scale model and designed two modules, the *Global Information Module (GIM)* and the *Local Information Module (LIM)*. The *GIM* used both convolutions and a *MSA* block while *LIM* is a convolution block. Moreover, they used the Inception-style method to learn multi-scale local representations. Their results show a great generalization capability where the model was generalizable to other cancer histopathological image classification tasks.

2.4.2. Registration

Recently ConvNets were used to solve the registration problem, but it was shown that convolution networks couldn't model long-range spatial dependencies very well. Transformers solved this issue due to self-attention, enabling more spatial precision in feature mapping. Inspired by *TransUNet* (Chen et al., 2021d), Chen et al. (2021c) used a hybrid of Convolutions and Transformer blocks in a VNet style where the encoder uses convolution blocks for feature extraction and in the decoder for the upsampling. In contrast, the bottleneck used the transformer blocks. In addition, the network used the whole 3D volume benefiting from the spatial information in the 3D format, which is very common in medical imaging. Chen et al. (2021b) used the same principle as *VoxelMorph* (Balakrishnan et al., 2019) where the network is learning to produce a dense displacement field between the fixed and moving image. *Swin* Liu et al. (2021) as an encoder for feature extraction, they used a convolution-based decoder with long skip connections to maintain the flow of spatial information for better registration. In addition, they used 3D convolution for more spatially aware features. (Zhang et al., 2021b) used the shifted-window self-attention layer from *Swin* (Liu et al., 2021). Each 3D volume is split into 3D patches then a convolution-based encoder and decoder are used to generate a deformation field between the fixed and moving patches. After computing the patch-wise deformation fields, *Swin* layer is used to stitch them, producing a whole deformation field for the entire volumes.

2.4.3. Object Detection

Shen et al. (2021) presented *COTR* architecture based on *DETR* (Carion et al., 2020) for polyp lesions detec-
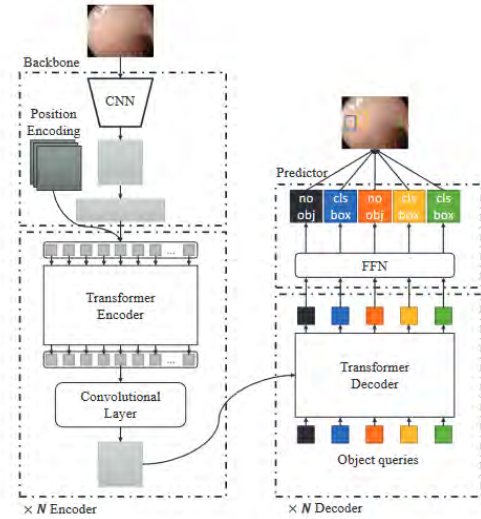
Figure 9: Shen et al. (2021), Overview of *COTR* architecture with the *convolution-in-transformer module*

Figure 10: Sun et al. (2021), Encoder architecture with two encoding paths, convolution-based and transformer-based

Figure 11: Zhang et al. (2021a), Overview of the TransFuse architecture, the transformer encoder is shown in the left part of the image while the convolution encoder is shown in the bottom right. The BiFusion module uses the output of the encoders per spatial level computing the final feature map for each level

tion in colonoscopies. The architecture uses *ResNet18* convolution backbone to extract high-level features. The features are fed into *convolution-in-transformer*, shown in figure 9 which is a block containing a self-attention layer followed by a convolution. The decoder follows the standard architecture for a transformer decoder except that the decoder uses the objects as a query running in parallel following *DETR*. A feed-forward network then uses the decoder output for the object classification and another one for the bounding box regression.

Jiang et al. (2021) presented *RDFNet* for caries detection, it is based on *YOLO V5* but they are using self-attention blocks with convolution layers.

2.4.4. Segmentation

Segmentation is an essential application in the medical imaging domain. With the emergence of vision transformers, many transformer-based architectures were designed and utilized for different segmentation tasks, for example, in cardiac segmentation, Chen et al. (2021d), Xu et al. (2021), Gao et al. (2021b), Zhou et al. (2021), and Cao et al. (2021) achieved a a state of the art results on different related datasets, while in multi-organ segmentation Chen et al. (2021d), Xu et al. (2021), Zhou et al. (2021), Chang et al. (2021), Li et al. (2021), Xie et al. (2021) and Cao et al. (2021) also achieved state of the art results. The same can be said many about many other tasks.

UNet-based architectures have achieved great success in the medical imaging domain, and many architectures are trying to utilize the transformer modules within the *UNet* architecture. The research into this idea can be categorized as follows. Strategies for combining transformer modules with the existing UNet architec-

ture and strategies to integrate transformer layers in existing UNet architecture directly.

Combining transformers with existing UNet *architectures:.* Several contributions try to merge the transformer modules with the existing UNet architecture without changing the UNet itself, Sun et al. (2021), used two independent encoding paths; the first is the standard convolution-based encoder used in UNet, while the other is a standard transformer-based encoder. Figure 10 shows the merging of their final outputs by a convolution block generating the last feature map. Zhang et al. (2021a), used the same strategy but added an extra fusion module, *BiFusion*, shown in figure 11 to combine the features from both encoders at each spatial level to leverage both the coarse and fine features maps extracted at each encoding level. The *BiFusion* uses convolution layers with attention to compute the final feature map.

Transformer UNet integration. Different contributions tried to integrate transformer layers directly into the architecture. One of the first approaches was to insert transformer layers as a bottleneck between the encoder and decoder, Chen et al. (2021d) presented *TransUNet*, shown in figure 12 with this integration strategy, The architecture used a standard convolution-based encoder for feature extraction followed by twelve transformer layers to extract better global information from the final feature map. The decoder is a standard UNet decoder
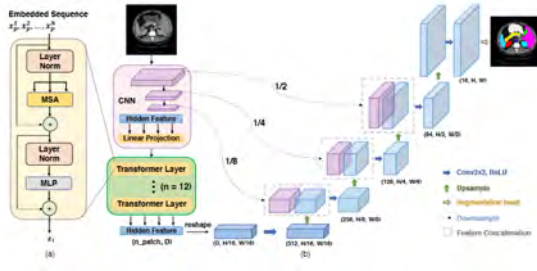
Figure 12: Chen et al. (2021d), Overview of the TransUNet architecture, (a) Schematic of the transformer layer, (b) Schematic of the network
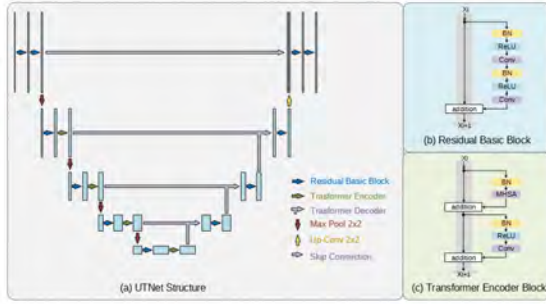
Figure 13: Gao et al. (2021b), (a) Overview of the UTNet architecture, (b) Pre-activation residual block, (c) Structure of the transformer encoder block

with the features extracted from the transformer bottleneck as the input. The architecture proved successful, beating many convolution-based networks.

Chang et al. (2021) did the same as *TransUNet* but integrated the transformer layers into the bottleneck of *Claw UNet* and achieved better results than *TransUNet* in multi-organ segmentation task. Xu et al. (2021) proposed *LeViT-UNet*, the novelty was in using the *LeViT* transformer.

Other approaches tried to integrate the transformer layers within the modules themselves, Li et al. (2021) used the transformer layer in the decoder for upsampling, they claim that using attention-based methods for the upsampling step in the decoder produces a significant difference compared to interpolation or deconvolutions.

Gao et al. (2021b) integrates the self-attention within each block of the network, figure 13. The encoder uses convolution-based residual blocks and passes the feature map to a transformer layer.

2.5. Data Efficiency in Vision Transformers

Transformer-based architectures have proven to achieve a state-of-the-art performance beating many convolution-based architectures, but these networks require a large amount of data. Some contributions start by pretraining the models on huge datasets and fine-tuning the trained model for the required task. For example, *Swin* (Liu et al., 2021) is a state-of-art model

for many tasks in computer vision. In classification it achieved state-of-art results on *ImageNet-1K* beating many convolution-based networks. They presented an ablation study showing the difference in performance between training only on *ImageNet-1K* vs pretraining on *ImageNet-21K* first then fine-tuning on *ImageNet-1K*. Similarly, *ViT* compared the results of pretraining on datasets with different magnitudes in size, *ImageNet-1K*, *ImageNet-21K*, *JFT-300M*. They showed the increased performance on the fine-tuned task with the increase in the size of the pretraining datasets.

Transformers demands larger amount of data relative to CNNs for training or for pertaining, which presents an obstacle in applying transformers in a lot of vision tasks.

3. Material and methods

In this section, we will discuss and explain the different approaches and experiments to tackle the task at hand. In section 3.1, we discuss the motivation of our approach. From section 3.2 to section 3.10 we discuss the different components utilized during this thesis and the final pipeline is summarized in section 3.11.

3.1. Motivation

CNNs and their variants have achieved state-of-the-art results in the problem of cardiac segmentation partially thanks to their progressively enlarged receptive fields that can learn a hierarchical feature representation. However, the long-term dependencies within images, such as the non-local correlation of objects in the image, are neglected in CNNs. For complex problems such as Infarction segmentation, that correlation is essential for comparing the different tissue types to detect anomalies.

Inspired by the success of transformers in computer vision tasks in general and medical imaging tasks specifically, we investigate the usage of the state-of-the-art transformer models in the application of cardiac image segmentation and then present our novel method.

3.2. Nested Hierarchical Transformer

In this section, A novel segmentation method will be presented based on the *Nested Hierarchical Transformer (NesT)* Zhang et al. (2022) architecture.

CNNs own their success for many reasons, mainly, The ability to extract hierarchical features from the image due to their increasing receptive field by the usage of different pooling techniques. And a strong inductive bias such as the locality of feature extraction or the pooling that focuses on the neighborhood of the pixel relative to its neighbors.

These gave a considerable edge to CNNs in computer vision tasks and led to more efficient, robust models to be trained with less data and less computation power.
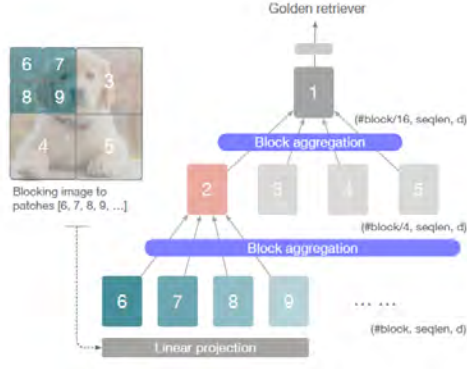
Figure 14: Zhang et al. (2022), overview of the NesT architecture

Figure 15: NesT Level, (a) Stack of *N* transformer layers composing the feature extractor (b) The aggregation block used for merging the extracted features

Although transformers exhibit a more substantial modeling power than CNNs, they are very data-hungry due to their lack of biases in the CNNs by design.

Their paper explored the idea of nesting basic local transformers on non-overlapping image blocks and hierarchically aggregating them. They found that the block aggregation function is critical in enabling cross-block non-local information communication. Based on this information, they introduced a simplified architecture compared to many transformers architectures such as *Swin*. *NesT* tries to incorporate both self-attention and convolution layers into one architecture playing to both of their strengths.

Figure 14 presents the high-level concept of the architecture. The architecture is a stack of levels to form the hierarchical representation of the image. Each level comprises a stack of transformer layers for feature extraction, followed by an aggregation block that uses the computed feature maps to merge the neighboring blocks into a single block with more refined features. The exact process is applied for every level giving the hierarchical structure presented in figure 14.

3.2.1. Embedding

The first step is to split the image into patches similar to other transformer networks. Given an input image of shape $H \times W \times C$, and a patch size of $S \times S$. The image is split into $S \times S$ patches then each patch will be linearly projected into an embedding vector in $\mathbb{R}^d$. The final number of patches will be

$$P_n = H \times W / S^2 \qquad (4)$$

Then all the embeddings are partitioned into blocks and flattened to generate the input $X \in \mathbb{R}^{B_n \times n \times d}$ where B_n is the number of blocks at the lowest NesT level, n is the sequence length (the number of patches) at each block and d is the embedding dimension of each patch. both B_n and n can be considered hyper-parameters of the network but they must satisfy the equality in equation 5 so the extracted patches can be split evenly into blocks

$$B_n \times n = P_n \qquad (5)$$
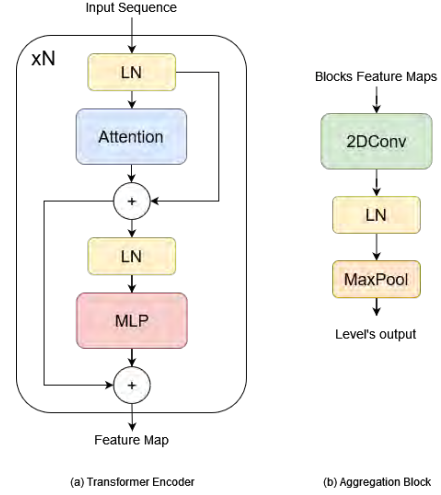
3.2.2. NesT Level

Each level is composed of two main components, the first, is a feature extraction module and the second is an aggregation module. The goal of the feature extractor is to compute feature map per input block while the aggregation block merge the output blocks' feature maps to ensure information sharing across different blocks and down-sample the spatial dimension by merging the neighbouring blocks into one.

Feature extractor:. Figure 15 (a) presents the schematics of the feature extractor. Each encoder is a stack of transformer layers following the design of Vaswani et al. (2017). Each layer is composed of multi-head self-attention *(MSA)* layer with the difference that the *MSA* process all the blocks independently and in parallel following equation 6.

$$\text{MSA}_{NesT}(Image) = \text{Stack}(F_1, F_2, \cdots, F_{T_n})$$
$$\text{where } F_i = \text{MSA}(block_i) \qquad (6)$$

MSA_{NesT} is followed by a feed-forward fully connected network *(FFN)* with skip connection and layer normalization *(LN)* (Ba et al., 2016). The architecture uses trainable position embeddings per transformer layer Touvron et al. (2021).

Block Aggregation. Local self-attention methods are essential for data efficiency and for reducing the computational requirements but in turn, it affects the information flow between the neighbouring blocks also the translation equivariance (Vaswani et al., 2021) of the network.

Swin Liu et al. (2021) achieves this task by shifting the block partition windows between each consecutive self-attention layer to connect adjacent blocks which is

very hard to implement. Also, *Swin* has to apply a special masked self-attention to guarantee spatial continuity due to the shifting adding extra complexity to the process.

NesT on the other hand uses simple convolution-pooling module, figure 15 (b) for information integrating between different blocks to compute finer, more global information. It is vital to apply the aggregation on the image plane so that the information can be exchanged between spatially close blocks. Also this induces extra bias from the shape of the image where the information exchange happens between neighbouring patches and blocks not across lowly correlated parts of the image. Figure 16 illustrates the advantage of aggregating the feature at the image plane to insure the exchange of information across the boundaries of neighbouring blocks.

3.2.3. *NesT, Experiments and Results*

The strength point of this architecture is data efficiency. To test their architecture they applied it to classification tasks on *CIFAR* (Krizhevsky et al., 2009) which is considered a small dataset for classification tasks, *ImageNet* 2012 benchmark Deng et al. (2009), and *ImageNet-21K* which are much bigger datasets. *NesT* models, other transformer models and CNNs are trained on these datasets from scratch with the CNNs acting as the baseline to compare the performance of the different transformers architectures.

ImageNet results are shown in table 2, *ImageNet* is considered a big dataset and it is used extensively by different CNN classification models which in turn been used either as feature extractors or for transfer learning but in table 2, convolution networks achieved similar results to different transformer methods either the ones that utilizes global self-attention, meaning that transformer networks can achieve results similar to CNN networks.

Local attention methods performs better compared to both CNNs and global attention transformers. on the other hand *NesT* models performs better than *Swin* models proving that even while using similar transformer layers, the aggregation of the different blocks plays vital role in dictating the output performance.

Finally, table 3 shows ImageNet results but with ImageNet-22K pretraining and as expected the overall performance of transformer networks increased but still *NesT-B* proved superior to other transformer models

3.3. *NesT for Segmentation*

Proving its ability to achieve state-of-art results with only training on relatively small datasets compared to other transformer networks. In this section, *NesT*-based models will be used for segmentation, as far as we know this is a novel application for this architecture.

Inspired by successful segmentation networks such as *TransFuse* (Zhang et al., 2021a) and *Swin-UNet* Cao

Arch. Type	Method	# Parameters	Top-1 Acc. (%)
	ResNet-50	25M	76.2
Convolution	RegNet Y-4G	21M	80.0
	RegNet Y-16G	84M	82.9
	ViT-B/16	86M	77.9
Global Attention	DeiT-S	22M	79.8
	DeiT-B	86M	81.8
	Swin-T	29M	81.3
	Swin-S	50M	83.3
Local Attention	Swin-B	88M	83.3
	NesT-T	17M	81.5
	NesT-S	38M	83.3
	NesT-B	68M	**83.9**

Table 2: Zhang et al. (2022), Comparison on the ImageNet dataset. All models are trained from random initialization.

	ViT-B/16	Swin-B	NesT-B
ImageNet Acc. (%)	84.0	86.0	**86.2**

Table 3: Zhang et al. (2022), Comparison on the ImageNet dataset results with ImageNet-22K pretraining

et al. (2021), we will use *NesT* as an encoder with pure *UNet* convolution-based decoder. To establish the baseline, we use the models provided by the authors, namely *NesT-T*, *NesT-S*, and *NesT-B* as encoders. All the variations of *NesT* have the same number of levels and same patch size, (4×4) but vary in other parameters, the differences are presented in table 4. We also use the pretrained weights, (trained on ImageNet from random initialization) provided by the authors. It is worth noting that *NesT-T* wasn't used because after some preliminary testing it produced inferior results to *NesT-S/B*

	Transformer Depth	Embedding Dimension	MSA number of Heads
NesT-T	[2, 2, 8]	[96, 192, 384]	[3, 6, 12]
NesT-S	[2, 2, 20]	[96, 192, 384]	[3, 6, 12]
NesT-B	[2, 2, 20]	[128, 256, 512]	[4, 8, 16]

Table 4: Comparison between the NesT architecture variations, transformer depth: how many transformer layers are stacked per NesT level, Embedding dimension: Number of feature channels per NesT level, and MSA number of Heads: The number of heads for the self-attention mechanism

3.3.1. *NesT-UNet Variations*

The first variation *NesT-UNet*, presented in figure 17a. A UNet architecture is employed where the encoder used is as-is *NesT* architecture with patch size of 4×4, although this cases a problem that the spatial reduction in size is not consistent with UNet style of halving the spatial size of the image in each encoder block. In *NesT*, an image of shape $H \times W$ is reduced to $H/4 \times W/4$ by the patch embedding and the first *NesT* layer. The Decoder used is a simple *2D Transposed Convolution* for upsampling followed by two convolution blocks. It is worth noting that using the pretrained weights, we needed to resize the input image to 224×224.

The second variation, *NesT-V2-UNet* is trying to increase the depth of the architecture and use 2×2 patch

Figure 16: Zhang et al. (2022), Illustration of block aggregation and a comparison when applying to the block plane versus on the image plane. Although both perform convolution and pooling spatially, performing block aggregation on the image plane allows information communication among blocks (different color palettes) that belong to different merged blocks at the upper hierarchy

size to create a better spatial features with more skip connections. The architecture presented in figure 17b. The pretrained weights are used as an initialization for some of levels of the network for better performance and faster convergence.

The third variation, *NesT-Dense-UNet* is trying to capitalize more on the spatial information extracted by the encoder while maintaining the high level features extracted by the original encoder with patch size 4×4 so instead an extra embedding layer is used in addition to the original one with patch size of 2×2 followed by a *NesT* level. Also a convolution encoder blocks used to extract low level features from the original image that can be used by the decoder to extract better features. The architecture is presented in figure 17c.

3.4. NesT as a Decoder

In this section we will present a novel decoder based on the *NesT* architecture. *NesT* encoder has proved that it can extract strong features suitable for medical image segmentation as shown in the results, section 5. As a next step we utilize the same principles of the architecture to create a decoder but instead of *block aggregation* module between each level, we implement *Block Expansion* module to increase the spatial dimensions after each *NesT* level. The architecture of the new block is presented in figure 18, The *Block Expansion* module consists of two steps, the first is the upsample of the image to twice its spatial size using *Bilinear Interpolation* followed by convolution layer for feature aggregation across patches and blocks. The second step is feature projection to reduce the number of channels, it starts by concatenating the newly computed features with the output of the encoder with the same spatial dimension and final *MLP* with the reduced number of channels. The main goal of the second step is to project the feature map into a lower dimension because the transformer block is not designed to change the number of the features shape.

3.5. Segmentation Losses

The choice of the loss functions are critical for any deep learning problem. For our knowledge there is no systematic way to choose the best loss function but recent evidence suggests that a combination of loss functions are the best for achieving state-of-the-art results. Iantsen et al. (2020) used a combination of *Dice Loss* and *Focal Loss* to achieve state-of-the-art for head and neck tumor segmentation while (Ma, 2021) used a combination of *Dice Loss* and *TopK Loss*. We utilize well established loss functions in this work and explore methods of combining them together to achieve higher performance.

Cross Entropy (CE):. Any segmentation problem can be modeled as a classification problem where each pixel is considered independent multi-class classification problem following equation 7 where each C is the number of possible classes for a pixel, G is the ground truth segmentation mask, and P is the protected probability by the model. A pixel-wise *CE* has been proven to work well in segmentation tasks.

$$CE = -\sum_{i}^{C} G_i \log P_i \qquad (7)$$

Dice Loss (DSC-L):. Dice coefficient is used to measure the overlap between the predicted segmentation and the ground truth so it can be used as an evaluation metric for segmentation. It has also been used as a loss function by inverting the metric into a loss function following equation 8. The dice score is explained in section 4.3 and in equation 12.

$$DSC\text{-}L = 1 - DSC(G, P) \qquad (8)$$

Segmentation Classification Loss (Seg-Cls):. Both the Infarction and No-Reflow classes don't exist in every slice or every volume meaning that the miss-segmentation of a single pixel will result in a very low evaluation score. To help solving this issue, we present a loss function based on the predicted segmentation map.

(a) UNet style architecture utilizing the NesT architecture as the encoder. Each decoder block is composed of *2D Transposed Convolution* followed by concatenation of the upsampled features with the encoder's skip connection features then finally two convolution block to compute the final output

(b) Enhancement on the first *NesT-UNet* by using a finer patch size 2×2

(c) Enhancement on the first *NesT-UNet* by using an extra two encoder block, the first is a NesT level with finer embedding patch size of 2×2 and the second is a convolution encoder block for very low feature extraction. Both are in addition to the original encoder for more tuned low level feature extraction.

Figure 17: The difference NesT variations used in the thesis
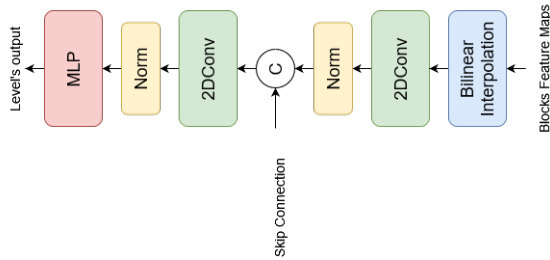
Figure 18: Block Expansion module used in the NesT decoder. It composes mainly of two steps, The upsampling step to increase the spatial resolution followed by projecting the features into a lower dimension

The algorithm is shown in algorithm 1. Focal loss (Lin et al., 2017) is known to handle imbalanced classes more efficiently by having a higher loss for poorly classified elements.

Compound Losses:. combing different losses have proven effective in many works so we presented different losses with different properties, *CE* is a pixel-wise loss function, *DSC-L* is a region-based loss working on a higher level than *CE*. It aims to maximize the overlap between entire regions not just on the pixel level. *HD-L* is a boundary loss which aims to minimize the distance between the components of the same class. Also, we presented a novel custom loss for slice classification minimizing the segmentation miss-classification

---

Algorithm 1 Segmentation Classification Loss

Require: P: Predicted Segmentation Probabilities, G: Ground Truth Mask

Loss = 0

for $c \in [\text{Infarction}, \text{No-Reflow}]$ **do**

    //G-label: Ground truth class label

    G-label = 1 *If* $c \in G$ *else* 0

    //P-probability: The Segmentation confidence in the existence of class c in the prediction

    P-probability = ***Non-Zero Mean***(P(c))

    Loss += Focal Loss(P-probability, G-label)

end for

// Returns the two classes average loss

return Loss / 2

// $\hat{y}$: probability of the predicted class $\in [0, 1]$

// y: Binary classification label $\in \{0, 1\}$

procedure FOCAL LOSS($\hat{y}$, y)

$$loss = \begin{cases} \alpha(1 - \hat{y})^{\gamma} \log \hat{y} & y = 1 \\ \alpha(\hat{y})^{\gamma} \log 1 - \hat{y} & y = 0 \end{cases}$$

    return loss

end procedure

---

rate. We explore the utilization of these different functions to enhance the overall results.

3.6. Preprocessing

EMIDEC dataset is composed of *LGE-MRI* scans for different patients. Each MRI consists of a stacked short-axis slices from base to apex of the left ventricle with the following features: pixel spacing between $1.25 \times 1.25mm^2$ and $2 \times 2mm^2$, slice thickness of $8mm$ and distance between slices between 8 and $13mm$. So as a preprocessing step, all the images are resampled into one spatial spatial spacing of $1.5 \times 1.5 \times 10mm$ based on the statistics of the training dataset.

Also to prevent the drawback of the displacement of the heart location between slices due to different breathholds, the slices are realigned according to the gravity center of the area defined by the epicardial contour. The specific positioning of the heart enabled us to process only specific part of the slices, a ROI of shape 96×96 is cropped.

The intensity range of the MRI scans is from 0 to more than 4000 so all the images are standardized, equation 9. The normalization helps the model to converge better, and faster.

$$S\,ample = \frac{S\,ample - Mean(S\,ample)}{S\,TD(S\,ample)} \qquad (9)$$

3.7. Data Augmentation

EMIDEC dataset is considered a small dataset with only 100 scans that are used for training and validation resulting in around 600 $2D$ slices for training. To help improve the quality of the training and avoid overfitting given the small dataset size we used different data augmentation techniques.

Random Rotation Augmentation:. The heart has specific geometrical circular shape, so to preserve that property we only used random rotation with a degree randomly sampled between $[-180, 180]$ with probability of 0.5.

Random Affine Augmentation:. We take the augmentation a step further with using a full affine transformation involving random translation, rotation, scaling and shearing.

Mix-Up:. The method used here is based on a proposed by Zhang et al. (2017) as an augmentation method for image classification. The methods original idea was to use pairs of training samples to generate a new one following equation 10 where I_1, I_2 are the image pair and y_1, y_2 are their corresponding labels. In this case the neural network trains on convex combinations of pairs of examples and their labels and by doing so, *Mix-Up* regularizes the neural network to favor simple linear behavior in-between training examples. Zhang et al.

(2017) proved that this approach increases the performance while reducing the overfitting of the model.

$$\begin{aligned} I_{new} &= \lambda I_1 + (1 - \lambda)I_2 \\ y_{new} &= \lambda y_1 + (1 - \lambda)y_2 \end{aligned} \qquad (10)$$

The problem at hand is segmentation not classification problem but the No-Reflow class is highly imbalanced so as a form of augmentation we utilized the principles of *Mix-Up* to generate more slices with the low frequency classes. The negative slices are blended with the other slices containing the low frequency class following equation 11 where I_P, I_N are the positive image slice and negative image slice respectively, and M_P, M_N are their corresponding masks. C is the set of low frequency classes, namely the Infarction and No-Reflow.

$$\begin{aligned} I_{new}(x, y) &= \begin{cases} \lambda * I_P(x, y) & M_P(x, y) \in C \\ (1 - \lambda) * I_N(x, y) & else \end{cases} \\ M_{new}(x, y) &= \begin{cases} M_P(x, y) & M_P(x, y) \in C \\ M_N(x, y) & else \end{cases} \end{aligned} \qquad (11)$$

Other works utilized similar techniques utilized *Mix-Up* for segmentation but in a different manor. Eaton-Rosen et al. (2018) used the original *Mix-Up* method but on the pixel level. They also apply selective process for choosing which patches to apply the augmentation to based on the foreground pixels' percentage. To our knowledge there is no method applying *Mix-Up* in this manor.

3.8. Postprocessing

The output of the *NesT UNets* provide very acceptable results in terms of preserving the anatomical shape properties of the Myocardium but in some cases the small regions are miss-classified as Infarction due to artifacts in the original image, these artifacts sometimes also magnified by the upsampling of the original image from 96×96 to 224×224 which is required by the architecture, to solve this issue we apply a postprocessing step on the output of the $2D$ networks. Figure 19 presents some examples before and after the postprocessing supporting the artifacts hypothesis. we compute the connected components for the Infarction class and use a threshold based on the area of the connected components to replace the infarction with Myocardium instead.

3.9. Pretraining

Transformers' performance has proven to scale with the size of the dataset used but different works tried to enhance their performance by pretraining the models on different tasks using supervised learning such as the original *ViT* (Dosovitskiy et al., 2020) and *Swin* (Liu
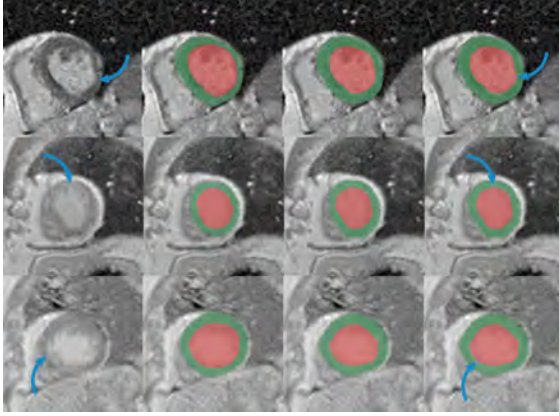
Figure 19: Examples of applying postprocessing on the models' output. The first column is the original image, the second is the ground truth mask, the third column is the postprocessed image and the fourth is the model's output. We notice in the first two images that the Infarction (purple) exists in only few pixels and they only appear on the boarder between the Myocardium (Green) and Left ventricle (Red) supporting the hypothesis that the miss-classification is due to artifacts in the original image.

et al., 2021). The usage of supervised training presents an obstacle in medical imaging due to the lack of supervised data. Many works tried to tackle the problem differently by using self-supervised learning techniques. Tang et al. (2021) introduced *UneTR*, which a *3D* segmentation network based on *Swin* architecture. In order to train their network, they introduced a novel self-supervised learning framework with task tailored for medical imaging analysis. They achieve state-of-the-art results on different segmentation datasets namely, *Beyond the Cranial Vault (BTCV)* Segmentation Challenge with 13 abdominal organs and segmentation tasks from the *Medical Segmentation Decathlon (MSD)* dataset (Antonelli et al., 2021). The only downside is that they utilized 5050 CT dataset for the pretraining. Zhou et al. (2022) introduce a simpler pretraining scheme for *ViT* compared to *UneTR*. They use *Masked Auto-Encoder (MAE)* with the task of reconstructing the original image from only partial observations. In this case the network is encouraged to learn the underlying structures of the images which is more relevant medical imaging compared to natural images processing. Medical imaging techniques are constrained by the anatomical structures in the image and by using self-supervision to reconstruct the original images, the network can learn these anatomical constrains improving the performance of downstream tasks. In this work, we utilize both self-supervision auto-encoder technique and a supervised pretraining and compare their performance.

Supervised:. *NesT* architectures are pretrained on the classification of *ImageNet* which is a nature images dataset, although it provide a boost for the performance of the network, the dataset has different properties compared to medical images. The goal of this experiment is to fine tune the network on a similar task before the final fine tuning on the required task. We chose the *ACDC* dataset, this dataset is part of a challenge to compare the performance of automatic methods on the segmentation of the left ventricular endocardium and epicardium as the right ventricular endocardium for both end diastolic and end systolic phase instances. Providing a very close application to the task at hand. The goal is to pretrain our architectures on the *ACDC* dataset before fine tuning on the *EMIDEC* dataset.

Self-Supervised:. Similar to the supervised pretraining approach, we are using a technique similar to *MAE* called in-painting. Each *2D* slice is sub-masked and the network is responsible for reconstructing the original image. We apply this method on both *ACDC* and *EMIDEC* datasets. Figure 21, presents some samples for the self-supervised inpainting task applied on the *EMIDEC* dataset.

For the self-supervised task, we added an extra branch to the architecture with the final encoder's feature map as its input. The new branch is composed of a cascade of deconvolutions to upscale the feature map and convolution layers for feature extraction and reduction. This task is optimized using the *L2* loss.

3.10. Interpretability

One of the main motivations behind the utilization of the *NesT* architectures is its inherit affinity to provide interpretable results. Interpretability, is a very hot topic in AI in general and in medical imaging specially. the nested hierarchy with the independent block process in *NesT* resembles a decision tree in which each block is encouraged to learn non-overlapping features and be selected by the block aggregation. This unique behavior motivated a new method for explaining the model reasoning, the authors presented *gradient-based class-aware tree-traversal (GradGAT)* method for classification interpretability.

The main idea is to find the most valuable traversal from a child node to the root node that contributes to the classification logits the most. Intuitively, at the top hierarchy, each of four child nodes processes one of 2×2 non-overlapping partitions of feature maps. We can use corresponding activation and class-specific gradient features to trace the high-value information flow recursively from the root to a leaf node. The negative gradient provides the gradient ascent direction to maximize the class c logit, meaning a higher positive value means higher importance.

Figure 20, presents an example of applying *GradGAT*. The *Radio telescope* image, the highest path is showing that the network is focusing on the right part of the image while on the other hand the *Lighter* image is showing that even if the model output is correct the model is not focusing fully on the right region in the image but it is focusing on the strong light source in the
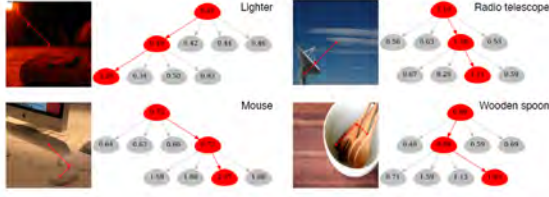
Figure 20: Zhang et al. (2022), Output visualization of the proposed *GradGAT*. Tree nodes annotate the averaged responses to the predicted class. We use a *NesT-S* with three tree hierarchies.

opposite corner to the lighter. We experimented with this method for interpreting the classification results of the *NesT* architecture.

3.11. Proposed Pipeline

In this section we will discuss the final pipeline and which techniques are used to compute the final results of the thesis. The final pipeline utilizes a *NesT-B UNet* for the segmentation task with the novel *NesT* based decoder. For preprocessing we implement simple random rotation data augmentation and for postprocessing we apply connected components removal based on area threshold. The segmentation network is pretrained on the *EMIDEC* dataset using a mixture of supervised task, namely the segmentation of said dataset and a self-supervised task, namely, inpainting on the same dataset.

4. Experiments Setup

4.1. Frameworks and Development Tools

All the experiments were done using *Pytorch* as the main deep learning framework. For *NesT* pretrained model we used *Pytorch Image Models* package (Wightman, 2019) which imports the original code and pretrained weights of the authors . Also, for segmentation utilities such as loss functions and different *UNet* based models, we used *Pytorch Segmentation Models* package Yakubovskiy (2020).

4.2. Training

In the development of the pipeline different number of architectures and their corresponding variances were trained, so we used an adaptive training process to accommodate for the different needs for each of the models. We used *AdamW* (Loshchilov and Hutter, 2017) optimizer with initial learning rate of 10^{-4}. *Reduce Learning Rate on Plateau* scheduler was used to adjust the learning rate based on the model performance. The scheduler reduces learning rate when a metric, in this case the validation loss has stopped improving. This gives some flexibility for the model to choose which learning rate range is more suitable for its training. All the models are trained for a maximum of 500 epochs with early stopping. The goal of early stopping is to

condition the termination of the training process on a metric achieving the best result while in process avoid overfitting. It is worth noting that some models such as *NesT-UNets* finished training in only $40 - 60$ epochs while the 3*D* models such as *NesT-UNet3D* required up to 300 epochs. Different models required different mini-batch sizes to fit in the GPU memory, So each model training starts with an estimation of the batch size and automatically each models finds the suitable batch size by trail and error. The hyperparameters were initially estimated by cross validation process. *AdamW* is producing better results compared to *SGD* and *Adam* optimizes. The optimizer used its default parameters, with 0.01 weight decay (L2 regularization), 0.9 as beta 1 and 0.999 as beta 2. All the *NesT* encoders trained from scratch initialize their weights with values drawn from a truncated normal distribution. The values are effectively drawn from the normal distribution with extreme values redrawn until they are within the acceptable bounds.

4.3. Evaluation

To evaluate the segmentation results, we used Dice coefficient to measure the similarity between the predicted segmentation and the ground truth. we also use the *Accuracy* to measure the model's performance in detecting the malignant tissue either the infarctions or the No-Reflow over the entire image. For example in the case of infarction accuracy, if there exist a pixel in the predicted mask with the infarction label, the label of the image is considered positive. The goal of this metric is to measure how well the segmentation network is detecting the malignant cases. These metrics are utilized for both 2*D* slices and 3*D* volumes.

Dice score follows equation 12, where *G* is the ground truth segmentation mask while *P* is the predicted mask. The accuracy measures the ratio between the correctly classified samples, namely true positives *(TP)* and true negatives *(TN)* and the overall number of samples following equation 13 when *FP* is the false positives and *FN* is the false negatives.

$$DSC(G, P) = \frac{2|G \cap P|}{|G| + |P|} \qquad (12)$$

$$Acc. = \frac{TP + TN}{TP + TN + FP + FN} \qquad (13)$$

To evaluate the overall performance of the models we used **five-fold cross validation** where the original hundred patience are split into eighty for train and twenty for validation. Then the final metric is computed based on the test dataset after the training. We report the average and the standard deviation of the five runs per experiment.

5. Results

5.1. Baseline

To establish a baseline for the new method, we use 2D UNet based methods which have established their success in the field, the first is a classical *UNet* and the second is *UNet-SE* which is a standard UNet with the addition of Squeeze & Excitation Blocks (Roy et al., 2018). In addition to different state-of-the-art transformer segmentation networks used successfully in the medical imaging domain namely, *TransUNet* (Chen et al., 2021d), *Swin-UNet* (Cao et al., 2021), and *Trans-Fuse* (Zhang et al., 2021a). To test the mentioned networks, they are trained from both random and with pretrained weights initializations. The results of training from scratch are presented in table 5, The convolution based methods such as *UNet*, *UNet-SE* perform better than *Swin UNet*, showing the ability of CNNs to perform better in the realm of small datasets. On the other hand, *TransUNet* is performing better than all of them because it capitalizes on the advantages of CNNs and transformers.

The same models were trained with their respected pretrained weights as initialization. The results of this experiments are presented in table 6. All the methods shows improvement in their overall performance but *Swin UNet* shows a significant improvement from the initialization compared to the other models and its own random initialization, testifying that pure transformer based models tend to require more data compared to other models and their performance increase with amount of data used, either for pretraining or actual training. Also, *Swin UNet* achieved results comparable to the CNN networks after lagging behind with random initialization. CNNs benefited from the initialization and the overall performance increased but they didn't surpass the *TransUNet*. *TransUNet* kept the best overall performance but *Swin UNet* scored much better in segmenting the infarctions.

5.2. NesT Variations

We compared the different NesT variations in two contexts same as the baseline models, the first is the training of the networks from scratch with random initialization and the second is using the *ImageNet* pretrained weights as initialization.

The results of the random initialization are presented in table 7. It is clear that almost all variations achieved better results than the pure transformer networks, *Swin UNet* and *UneTR* with almost 10 dice points in each category while achieving comparable results to the CNN networks, *UNet*, *UNet-SE* which proves the network ability to perform well in the range of small datasets compared to transformer networks. On the other hand, *TransUNet* proved better at learning that the *NesT* variations.

The results of using pretrained weights as initialization are presented in table 8. With the usage of the pretrained weight, *NesT* variations proved more robust compared to all the baseline models, including *TransUNet* especially in Myocardium, and Infarction segmentation but lacks a bit with the No-Reflow segmentation. It is also worth noting that *TransUNet* use the weights from pretraining on *ImageNet-21K* while *NesT* variations use weights from pretraining on only *ImageNet*. The overall performance of *NesT* is encouraging to push for better segmentation results.

5.3. Data Augmentation

Data augmentation is widely used for either enhancing the performance or avoiding overfitting due to small dataset size. 3.7 explains the augmentation techniques used through this experiment. All the expirments used *Cross Entropy and Dice Losses* and used the ImageNet weights as initialization. Table 9 presents the results for the classical augmentation techniques. Both *NesT-B Dense* and *NesT-B* performed much better with a simple augmentation,

Mix-Up is used to help with the Infarction, and No-Reflow class imbalance in the number of pixels per sample. Table 10, presents the results of the *Mix-Up* experiments. The technique didn't enhance the performance much compared to simple augmentations,

5.4. Segmentation Loss Functions

In this section, we compare the different combinations of loss function used. The experiment uses simple random rotation augmentation mentioned previously.

A combination of Cross Entropy Loss *(CE)* and Dice Loss *DSC-L* is considered a defacto in image segmentation and we consider it the baseline loss function used by all the experiments before. In addition we experimented with our novel loss function explained in section 3.5. Table 11, presents a comparison between the loss functions. It is clear that the addition of the novel loss function increased the Dice score for both the Infarction and No-Reflow classes. *Nest-B* increased by 2 dice points in 2D No-reflow and 4 in 3D segmentation, while the Infarction increased by 4 points in 2D and 2 points in 3D.

5.5. Pretraining

The performance of any model is dependant on the amount of data used for pretraining which has been proved in sections 5.1 and 5.2. In this section we discuss the results of using different datasets and different pretraining techniques. For this experiment we use the *Nest-B UNet* architecture, a combination of *CE*, *DSC-L*, and *Seg-Cls* losses, and random rotation augmentation.

Table 12 compares between the performance using the pretraining on both the *EMIDEC* and *ACDC*. For *ACDC* dataset, the results were similar to the results

NesT UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial Infarction Evaluation

17

Model	Average DSC		Mayo DSC		Infarction					No-Reflow			
	2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
UNet	69.63 ± 2.13	61.06 ± 3.11	76.47 ± 1.6	78.21 ± 1.56	47.92 ± 3.64	60.96 ± 2.04	43.22 ± 1.64	82.0 ± 3.16	85.36 ± 1.89	84.49 ± 3.02	61.76 ± 7.92	71.6 ± 2.19	86.15 ± 1.61
UNet-SE (Roy et al., 2018)	69.0 ± 1.94	60.88 ± 3.41	75.11 ± 0.86	76.93 ± 0.94	47.14 ± 4.94	58.35 ± 2.31	42.61 ± 3.07	83.2 ± 2.68	83.02 ± 1.98	84.74 ± 3.56	63.09 ± 9.35	72.8 ± 4.6	87.09 ± 1.67
TransUNet (Chen et al., 2021d)	71.64 ± 1.19	62.79 ± 2.43	80.2 ± 1.12	81.23 ± 1.17	51.19 ± 6.55	60.8 ± 4.38	46.42 ± 8.15	82.0 ± 7.62	84.02 ± 3.52	83.52 ± 5.09	60.72 ± 11.78	73.6 ± 6.69	86.48 ± 3.68
Swin UNet (Cao et al., 2021)	60.51 ± 5.66	51.45 ± 6.33	73.16 ± 1.44	74.6 ± 1.39	38.58 ± 5.72	56.27 ± 5.52	41.03 ± 5.12	82.8 ± 3.03	83.18 ± 2.26	69.78 ± 12.21	38.73 ± 13.29	55.6 ± 11.35	75.14 ± 11.07
UNETR2D (Tang et al., 2021)	60.53 ± 3.52	52.72 ± 8.08	77.05 ± 0.82	78.7 ± 0.82	36.41 ± 2.88	58.79 ± 2.34	42.16 ± 5.73	82.8 ± 3.9	85.98 ± 1.75	68.11 ± 12.51	37.3 ± 23.81	53.2 ± 15.4	74.3 ± 9.03

Table 5: Baseline methods trained from random initialization. *UNet-SE* is a standard UNet with the addition of Squeeze & Excitation Blocks (Roy et al., 2018).

Model	Average DSC		Mayo DSC		Infarction					No-Reflow			
	2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
UNet	74.49 ± 1.31	66.39 ± 2.65	81.71 ± 0.66	82.83 ± 0.63	58.2 ± 2.7	67.22 ± 1.45	54.85 ± 2.58	89.2 ± 3.03	88.83 ± 1.34	83.55 ± 2.68	61.49 ± 6.37	76.8 ± 6.26	88.32 ± 2.36
UNet-SE (Roy et al., 2018)	73.41 ± 1.53	63.0 ± 2.21	81.13 ± 1.04	82.26 ± 0.92	56.65 ± 2.34	65.11 ± 2.41	50.92 ± 3.65	86.4 ± 1.67	87.04 ± 2.43	82.46 ± 2.61	55.82 ± 4.94	70.4 ± 3.29	87.88 ± 3.2
TransUNet (Chen et al., 2021d)	75.93 ± 1.83	65.99 ± 3.79	82.14 ± 0.47	83.08 ± 0.45	59.39 ± 4.93	64.51 ± 2.51	50.83 ± 6.68	84.4 ± 6.07	85.47 ± 2.44	86.27 ± 1.46	64.05 ± 5.89	76.4 ± 4.34	89.44 ± 1.0
Swin UNet (Cao et al., 2021)	73.75 ± 4.05	64.73 ± 4.46	81.32 ± 0.89	82.42 ± 0.72	60.61 ± 2.82	67.92 ± 1.42	57.2 ± 4.71	89.2 ± 4.6	89.16 ± 1.03	79.33 ± 8.88	54.58 ± 10.21	70.8 ± 8.9	85.03 ± 6.81

Table 6: Baseline methods trained with pretrained weights initialization. *UNet-SE* is a standard UNet with the addition of Squeeze & Excitation Blocks Roy et al. (2018)

Model	Average DSC		Mayo DSC		Infarction					No-Reflow			
	2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
NesT-S UNet	66.72 ± 3.35	58.02 ± 6.21	78.1 ± 1.7	79.68 ± 1.72	51.66 ± 2.13	61.27 ± 1.53	47.96 ± 3.8	86.8 ± 2.28	86.31 ± 0.71	70.41 ± 11.17	46.42 ± 17.85	61.2 ± 9.86	75.64 ± 8.38
NesT-B UNet	68.76 ± 3.33	60.49 ± 3.75	78.51 ± 1.24	79.89 ± 1.33	52.76 ± 6.27	62.28 ± 2.63	51.15 ± 5.95	87.2 ± 4.15	86.03 ± 1.28	75.01 ± 2.91	50.43 ± 5.02	66.4 ± 4.1	80.61 ± 2.28
NesT-S V2 UNet	63.5 ± 6.9	53.71 ± 8.71	76.56 ± 1.48	78.1 ± 1.33	48.29 ± 5.35	57.53 ± 6.3	42.16 ± 7.64	86.0 ± 5.1	86.37 ± 2.0	65.66 ± 16.13	40.87 ± 18.83	56.4 ± 14.52	71.62 ± 14.42
NesT-B V2 UNet	68.89 ± 2.99	58.18 ± 4.36	76.91 ± 1.48	78.32 ± 1.53	53.7 ± 4.69	60.8 ± 4.13		84.8 ± 4.38	85.7 ± 2.18	76.06 ± 4.36	48.92 ± 7.03	66.0 ± 6.32	81.28 ± 3.85
NesT-S Dense UNet	66.58 ± 3.46	56.33 ± 6.44	78.4 ± 0.62	79.8 ± 0.61	51.44 ± 1.55	60.15 ± 1.59	45.93 ± 3.92	86.0 ± 1.41	86.15 ± 0.9	69.89 ± 11.63	43.25 ± 16.05	61.2 ± 15.21	76.98 ± 11.22
NesT-B Dense UNet	67.06 ± 3.7	57.48 ± 5.6	78.85 ± 0.94	80.38 ± 0.75	52.28 ± 1.72	60.56 ± 1.81	47.26 ± 4.04	85.2 ± 3.03	84.97 ± 0.89	70.05 ± 10.56	44.8 ± 13.53	62.8 ± 14.18	75.98 ± 9.85

Table 7: NesT UNets variations, Trained from random initialization

Model	Average DSC		Mayo DSC		Infarction					No-Reflow			
	2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
NesT-S UNet	69.51 ± 6.77	58.85 ± 8.14	83.54 ± 0.34	84.51 ± 0.34	58.39 ± 7.44	66.56 ± 5.46	53.35 ± 7.61	85.6 ± 5.55	89.11 ± 2.2	66.58 ± 13.18	38.67 ± 17.21	54.0 ± 15.03	72.01 ± 12.43
NesT-B UNet	71.5 ± 0.75	61.92 ± 1.49	83.93 ± 0.63	84.8 ± 0.58	59.6 ± 1.83	68.57 ± 1.29	55.6 ± 2.5	86.8 ± 3.03	88.77 ± 1.85	70.95 ± 1.89	45.38 ± 4.18	61.2 ± 4.38	76.26 ± 1.91
NesT-S V2 UNet	69.95 ± 3.64	60.41 ± 3.48	80.83 ± 1.27	81.99 ± 1.18	55.66 ± 6.35	64.04 ± 3.69	50.82 ± 5.28	85.2 ± 4.15	86.59 ± 2.26	73.36 ± 5.55	48.43 ± 6.12	64.8 ± 7.01	79.11 ± 5.52
NesT-B V2 UNet	72.46 ± 0.74	63.34 ± 1.15	81.24 ± 0.78	82.28 ± 0.67	58.69 ± 3.38	67.06 ± 1.0	54.03 ± 3.37	85.2 ± 3.9	88.49 ± 1.14	77.46 ± 1.13	53.72 ± 3.33	70.8 ± 4.38	82.57 ± 1.18
NesT-S Dense UNet	76.41 ± 2.58	65.76 ± 2.35	84.31 ± 0.32	85.18 ± 0.32	63.33 ± 4.4	69.75 ± 1.29	56.86 ± 2.35	86.4 ± 3.29	88.32 ± 0.85	81.58 ± 4.1	55.24 ± 6.24	69.2 ± 6.42	86.26 ± 4.26
NesT-B Dense UNet	73.12 ± 3.35	62.95 ± 3.64	84.13 ± 0.34	84.96 ± 0.32	60.87 ± 2.11	67.95 ± 1.55	54.8 ± 2.47	85.6 ± 3.29	88.04 ± 1.0	74.37 ± 9.42	49.07 ± 9.92	63.2 ± 8.44	79.61 ± 8.72

Table 8: NesT UNets variations, Trained with the pretrained weights as initialization

Method	Model	Average DSC		Mayo DSC		Infarction					No-Reflow			
		2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
Random Rotation	NesT-S UNet	79.36 ± 1.42	70.33 ± 1.38	85.47 ± 0.52	86.35 ± 0.52	67.24 ± 2.37	72.95 ± 0.29	62.81 ± 1.54	89.6 ± 0.89	90.11 ± 0.25	85.37 ± 2.32	61.84 ± 5.16	73.6 ± 5.18	89.16 ± 2.13
	NesT-B UNet	78.9 ± 2.04	68.5 ± 4.69	85.23 ± 0.4	86.13 ± 0.37	66.79 ± 2.74	72.1 ± 1.3	60.93 ± 3.7	90.0 ± 2.0	90.17 ± 0.98	84.69 ± 4.36	58.45 ± 11.32	71.6 ± 10.14	88.49 ± 3.67
	NesT-S V2 UNet	76.95 ± 1.26	67.96 ± 2.25	82.98 ± 0.69	84.08 ± 0.56	64.31 ± 3.0	69.76 ± 1.22	58.32 ± 1.57	88.8 ± 1.79	88.21 ± 1.87	83.55 ± 2.66	61.47 ± 5.98	73.6 ± 5.73	87.32 ± 2.69
	NesT-B V2 UNet	75.47 ± 3.32	66.61 ± 4.61	83.78 ± 0.92	84.67 ± 0.91	63.49 ± 2.95	70.09 ± 1.88	59.9 ± 2.97	89.2 ± 3.63	88.99 ± 0.98	79.13 ± 7.56	55.26 ± 12.73	69.2 ± 10.55	83.46 ± 6.86
	NesT-S Dense UNet	78.74 ± 2.42	69.28 ± 3.84	85.5 ± 0.67	86.33 ± 0.77	64.37 ± 5.19	71.94 ± 2.36	62.56 ± 4.57	88.8 ± 4.15	89.39 ± 0.86	86.34 ± 1.66	58.96 ± 6.66	70.4 ± 4.98	88.94 ± 2.77
	NesT-B Dense UNet	79.49 ± 0.75	70.44 ± 1.45	85.46 ± 0.27	86.27 ± 0.25	66.66 ± 1.34	72.51 ± 0.85	62.74 ± 0.65	90.4 ± 0.89	90.0 ± 1.6	86.35 ± 1.59	62.29 ± 4.3	73.2 ± 3.03	90.11 ± 1.58
Affine	NesT-S UNet	76.84 ± 3.27	67.23 ± 4.66	84.26 ± 0.64	85.24 ± 0.65	63.4 ± 1.92	71.33 ± 0.97	57.64 ± 3.46	86.4 ± 2.97	90.0 ± 0.31	82.87 ± 8.54	58.81 ± 10.56	76.8 ± 8.79	86.76 ± 8.04
	NesT-B UNet	78.14 ± 1.29	68.82 ± 2.39	83.99 ± 0.44	84.94 ± 0.5	63.3 ± 2.54	71.22 ± 0.68	59.62 ± 3.52	88.0 ± 3.74	89.83 ± 0.87	87.11 ± 2.19	61.89 ± 6.35	73.6 ± 4.56	90.61 ± 2.11
	NesT-S V2 UNet	74.13 ± 3.16	64.5 ± 4.58	82.37 ± 0.29	83.43 ± 0.3	59.69 ± 3.87	67.68 ± 1.69	54.07 ± 4.34	84.8 ± 3.63	87.77 ± 1.11	80.34 ± 7.03	56.01 ± 10.13	70.8 ± 9.23	85.14 ± 6.56
	NesT-B V2 UNet	74.8 ± 2.13	65.67 ± 2.72	82.28 ± 0.78	83.41 ± 0.77	60.62 ± 4.32	68.26 ± 2.29	56.61 ± 2.72	88.4 ± 2.19	87.21 ± 1.52	81.5 ± 4.03	56.99 ± 6.6	70.8 ± 6.72	86.2 ± 3.55
	NesT-S Dense UNet	76.23 ± 2.36	68.1 ± 2.11	84.16 ± 0.65	85.07 ± 0.59	60.17 ± 4.34	70.43 ± 1.51	59.05 ± 2.17	87.6 ± 2.19	89.66 ± 1.2	84.36 ± 2.98	60.18 ± 4.38	72.4 ± 3.85	88.1 ± 2.48
	NesT-B Dense UNet	78.72 ± 1.33	70.17 ± 2.38	84.4 ± 0.36	85.36 ± 0.35	65.58 ± 1.98	71.61 ± 0.99	60.68 ± 2.36	89.6 ± 2.61	89.89 ± 0.67	86.18 ± 2.2	64.46 ± 6.0	75.6 ± 5.55	89.72 ± 2.27

Table 9: Results of applying different Augmentation techniques on different model variations. Namely, Random rotation and Affine transformation

without the pretraining but *EMIDEC* pretraining proved effective in increasing the performance of the model especially for the *Segmentation and Inpaint* task at the same time, figure 21 presents some qualitative examples of both the reconstruction and segmentation during the pretraining with very good reconstruction results, especially for the underlying structures of the image, namely the Myocardium and very good segmentation results even with the sub-masked images.

5.6. NesT Decoder

We experimented with different architectures for the *Block Expansion* and we report the best results in table 13, The new decoder produce results comparable to the convolution decoder except for No-Reflow where the novel decoder produce higher results. The best convolution decoder evaluated at 70.34 average dice points on 3*D* No-Reflow segmentation while the novel *NesT* decoder evaluated at 76.52 average dice points.

5.7. Results Analysis based on Anatomical Features

In this section we discuss the performance of the trained models on the different anatomy of the heart. Figure 22 presents the difference in both size and thickness of the Myocardium and also presenting the challenge in processing both the Apex and Basal slices. Table 22, presents a comparison between the performance of different models on the different slice types.

6. Exploring GradGAT

As explained in section 3.10, *NesT* hierarchical tree structure is very useful for results interpretability and it proved useful on datasets such as *ImageNet* which was one of the reasons that encouraged us to work with this architecture. To test this method we implemented simple classification head following the same architecture as the *NesT* paper. The classification task was trained to classify if a slice has Infarction and No-Reflow tissue in a multi-label classification scheme.

Model	Mix-Up λ	Average DSC		Mayo DSC		Infarction					No-Reflow			
		2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
NesT-B	0.2	77.86 ± 1.07	68.1 ± 1.96	82.73 ± 0.74	83.85 ± 0.64	60.36 ± 2.7	66.13 ± 1.47	57.19 ± 1.97	86.0 ± 2.83	79.83 ± 2.21	84.73 ± 3.4	63.27 ± 4.73	75.6 ± 2.19	88.6 ± 3.01
UNet	0.8	77.56 ± 1.51	67.39 ± 2.56	82.51 ± 0.56	83.59 ± 0.56	57.63 ± 3.6	66.98 ± 2.13	59.57 ± 4.55	88.4 ± 5.18	80.89 ± 3.28	83.21 ± 3.4	59.0 ± 6.62	73.2 ± 3.03	87.21 ± 2.82
NesT-B	0.2	76.88 ± 1.33	66.8 ± 2.38	82.41 ± 0.52	83.37 ± 0.57	58.09 ± 2.52	65.83 ± 1.72	56.73 ± 4.41	86.8 ± 5.22	80.22 ± 1.43	82.39 ± 2.24	60.3 ± 4.03	75.2 ± 3.63	87.09 ± 1.6
Dense UNet	0.8	77.81 ± 2.55	66.94 ± 5.05	82.74 ± 1.0	83.77 ± 0.91	57.97 ± 5.32	66.61 ± 1.56	57.74 ± 4.38	88.8 ± 3.03	80.11 ± 2.18	84.08 ± 6.36	59.31 ± 11.25	70.8 ± 8.9	87.71 ± 5.4

Table 10: Results of applying the Mix-Up Augmentation with different λ on the *NesT-B UNet* model

Model	Loss	Average DSC		Mayo DSC		Infarction					No-Reflow			
		2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
NesT-B UNet	CE + DSC	78.7 ± 1.07	70.96 ± 1.02	85.1 ± 0.52	85.96 ± 0.52	64.05 ± 2.61	72.36 ± 1.48	62.01 ± 2.99	89.6 ± 3.29	90.84 ± 1.46	86.95 ± 2.09	64.9 ± 4.74	75.6 ± 5.9	90.22 ± 2.28
	CE + DSC + Seg-Cls	80.56 ± 1.03	73.51 ± 2.29	84.79 ± 0.49	85.65 ± 0.5	68.46 ± 1.68	72.64 ± 1.27	64.42 ± 2.3	92.8 ± 1.79	90.17 ± 0.87	88.42 ± 1.43	70.45 ± 4.59	80.4 ± 2.19	91.45 ± 0.96
NesT-B	CE + DSC	79.12 ± 1.36	70.44 ± 2.1	84.89 ± 1.03	85.75 ± 0.99	64.96 ± 2.93	71.94 ± 1.05	59.99 ± 2.33	86.8 ± 3.03	89.5 ± 0.8	87.52 ± 1.8	65.59 ± 4.76	77.2 ± 3.9	91.34 ± 1.49
Dense UNet	CE + DSC + Seg-Cls	80.27 ± 0.73	71.68 ± 1.55	85.01 ± 0.43	85.89 ± 0.45	69.19 ± 1.09	72.75 ± 0.74	63.6 ± 1.68	91.6 ± 1.67	90.17 ± 1.55	86.6 ± 1.03	65.54 ± 3.32	77.2 ± 3.03	90.56 ± 1.29

Table 11: Comparison between different loss functions and their combinations. Mainly the table shows the increase in performance with the addition of the novel *Seg-Cls* loss function

Dataset	Method	Average DSC		Mayo DSC		Infarction					No-Reflow			
		2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
ACDC	inpaint	78.42 ± 0.98	68.63 ± 1.32	83.81 ± 0.79	84.69 ± 0.64	65.89 ± 2.11	70.37 ± 0.75	59.76 ± 0.72	90.0 ± 2.45	89.78 ± 1.32	85.57 ± 1.69	61.44 ± 3.47	74.8 ± 3.63	89.78 ± 1.68
	seg	79.83 ± 1.09	71.5 ± 1.74	84.75 ± 0.43	85.64 ± 0.42	68.72 ± 0.52	72.65 ± 0.28	62.14 ± 2.36	89.33 ± 2.31	91.34 ± 0.28	86.03 ± 3.07	66.73 ± 5.73	78.0 ± 4.0	89.76 ± 2.64
	seg-inpaint	80.52 ± 1.1	71.39 ± 2.35	85.17 ± 0.34	86.1 ± 0.28	69.11 ± 1.89	71.94 ± 1.08	61.34 ± 3.31	89.33 ± 2.73	89.76 ± 0.52	87.27 ± 1.85	66.73 ± 4.3	78.33 ± 3.88	90.64 ± 1.94
EMIDEC	inpaint	80.7 ± 0.44	72.47 ± 1.07	84.92 ± 0.3	85.81 ± 0.29	68.13 ± 1.2	72.4 ± 1.03	62.51 ± 1.9	90.4 ± 2.19	89.83 ± 0.75	89.04 ± 1.08	69.1 ± 3.74	79.2 ± 2.28	92.74 ± 1.41
	seg-inpaint	82.12 ± 0.49	74.2 ± 0.93	85.83 ± 0.14	86.67 ± 0.16	71.54 ± 1.24	73.73 ± 0.97	65.59 ± 1.8	92.4 ± 1.67	90.61 ± 1.09	89.0 ± 0.99	70.34 ± 3.0	80.8 ± 3.03	92.07 ± 1.0

Table 12: Comparison between the different Pretraining techniques, inpainting, segmentation and inpaint-segmentation on both the *EMIDEC* and the *ACDC* dataset

Dataset	Method	Average DSC		Mayo DSC		Infarction					No-Reflow			
		2D DSC	3D DSC	2D DSC	3D DSC	2D DSC	Post 2D DSC	3D DSC	Case Acc.	Slice Acc.	2D DSC	3D DSC	Case Acc.	Slice Acc.
ACDC	inpaint	80.1 ± 1.18	72.19 ± 2.69	84.35 ± 1.2	85.26 ± 1.0	69.67 ± 1.78	72.49 ± 1.31	63.46 ± 2.14	92.0 ± 2.0	90.17 ± 0.96	86.28 ± 2.0	67.83 ± 5.95	81.6 ± 5.9	90.56 ± 1.93
	seg	80.71 ± 0.87	71.88 ± 1.5	84.65 ± 0.42	85.58 ± 0.35	70.27 ± 1.85	72.63 ± 1.13	62.88 ± 2.24	90.0 ± 2.45	89.83 ± 0.9	87.19 ± 0.95	67.17 ± 2.5	78.4 ± 3.58	90.56 ± 0.82
	seg-inpaint	81.01 ± 1.37	71.84 ± 1.94	85.45 ± 0.63	86.36 ± 0.53	71.43 ± 1.45	73.91 ± 1.96	64.46 ± 2.73	91.6 ± 1.67	91.45 ± 1.73	86.15 ± 2.34	64.7 ± 3.04	76.4 ± 3.29	89.78 ± 1.97
EMIDEC	inpaint	78.52 ± 0.72	70.21 ± 1.31	83.08 ± 1.4	84.07 ± 1.31	67.22 ± 1.06	70.34 ± 1.12	62.52 ± 1.82	92.8 ± 2.28	88.83 ± 1.35	85.26 ± 0.85	64.03 ± 2.91	76.8 ± 2.28	88.88 ± 0.41
	seg-inpaint	82.53 ± 0.74	75.6 ± 1.05	85.28 ± 0.21	86.18 ± 0.22	71.68 ± 1.73	72.55 ± 1.09	64.11 ± 1.45	92.0 ± 2.83	88.66 ± 1.87	90.64 ± 0.49	76.52 ± 2.43	86.4 ± 3.29	93.3 ± 1.19

Table 13: Results of implementing the *NesT* decoder instead of the standard convolution decoder used by *UNet*
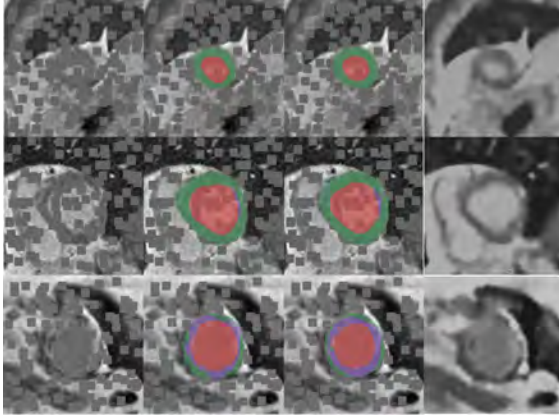
Figure 21: Examples of Segmentation-inpaint pretraining using the *EMIDEC* dataset. The first column is the sub-masked image, the second column is the ground truth mask, the third is the model's segmentation output and the fourth is the reconstructed images. It is clear that the reconstruction process is successful in generating the underlying structures of the image, namely the Myocardium and Left-ventricle regions. Also it shows the exceptional results from the segmentation giving the missing information of the image.

Figure 22: The figure presents the difference between slices based on their location in the MRI sequence. The left image is the Apex which is smallest area in the sequence which makes it very hard to process, the middle is a Middle slice, they are the easiest to process and the right is Basal, the main reason for their difficulty is the existence of multiple structures around the cavity such as the Right-Ventricle

From our expirments, the method is not suitable for all classification tasks especially for complex dataset such as *EMIDEC*. The aggregation of the activation values across the different levels leads to misleading results due to its over simplicity. In the current method the highest level will compute the average activation map using a 2×2 grid choosing the best quadrant, meaning the quadrant contributing the highest to the final score,

followed by the same process on the selected quadrant. This simple aggregation leads to bad results when the region of interest is at the boarder between the quadrants or the ROI exists in two quadrants at once. In figure 23, the *GradGAT* managed to get close to the best ROI in the image but the aggregation failed to hone in the right one and of course it completely missed the Infarction in the bottom right quadrant of the image.

Although the method wasn't effective for our case but it presents an interesting hierarchical method that can be used for interpretability of the results. In future work, a good approach instead of using the current aggregation method, we can look at the tree as a whole and use different techniques from graph theory such as *Min-Cut*.

7. Discussion

In this work we implemented novel segmentation network based on the *NesT* architecture to establish a fair baseline we utilized well established methodologies in

Model	Slice Location	Myo 2D DSC	Infarction 2D DSC	No-Reflow 2D Dice
	Apex	74.52 ± 1.09	61.0 ± 1.81	86.02 ± 2.17
UNet	Middle	82.56 ± 0.72	67.08 ± 1.38	80.62 ± 2.92
	Basal	84.51 ± 0.85	74.21 ± 5.42	96.21 ± 3.42
	Apex	73.74 ± 1.97	58.61 ± 5.34	79.83 ± 3.83
UNet-SE	Middle	82.08 ± 1.05	65.32 ± 2.22	80.79 ± 3.32
	Basal	83.61 ± 0.65	70.5 ± 2.19	93.74 ± 2.01
	Apex	74.88 ± 0.77	59.44 ± 2.14	86.61 ± 1.45
TransUNet	Middle	83.18 ± 0.62	64.37 ± 2.08	84.31 ± 1.58
	Basal	84.05 ± 0.97	70.32 ± 9.23	96.05 ± 4.5
	Apex	74.78 ± 1.18	62.6 ± 2.86	81.74 ± 6.98
Swin UNet	Middle	82.09 ± 0.93	67.89 ± 1.81	76.9 ± 9.29
	Basal	83.88 ± 1.11	73.36 ± 1.92	89.45 ± 9.45
	Apex	80.67 ± 0.55	70.73 ± 1.41	87.17 ± 2.94
Nest-B + Conv Decoder	Middle	86.38 ± 0.26	72.6 ± 1.15	87.61 ± 1.55
	Basal	88.14 ± 0.18	82.57 ± 2.14	98.02 ± 0.04
	Apex	79.51 ± 0.81	68.52 ± 2.84	88.44 ± 1.64
Nest-B + NesT Decoder	Middle	85.92 ± 0.25	72.09 ± 0.79	89.69 ± 0.49
	Basal	87.74 ± 0.11	78.93 ± 1.72	97.71 ± 0.99

Table 14: Analysis of the results based on the autonomy of the slice where the Apex is the smallest slice in the MRI, the Basal which is the top slice and the rest are the Middle slices
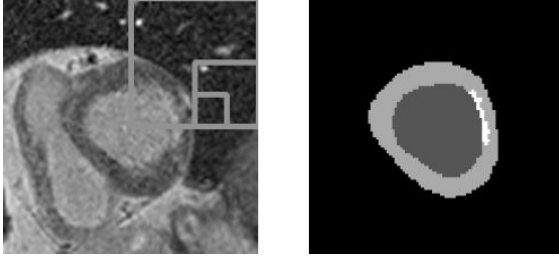

Figure 23: The Left image presents the selected quadrants one inside the other, we can clearly see that the final quadrant is very near to the Infarction. The right figure is the ground truth mask where the white color represents the infarcted tissue.

CNNs and transformer segmentation networks. From the results at section 5.1, we can conclude that CNNs still perform better than other networks when trained on small datasets but at the same time from the literature review we conclude that enhancing their performance is becoming increasingly difficult. and requires multiple networks performing multiple tasks to enhance the results. *ICPIU-Net* utilizes two different networks with multiple different losses and sub-tasks also it was training using a complex training procedure while (Zhang, 2020) followed the same design as *nnUNet* utilized a 2*D* UNet followed 3*D* Unet. Another direction integrated CNNs with transformer modules, namely *TransUNet* and achieved great performance improvement utilizing the best of both worlds but this shows the limited capacity of CNNs to model difficult problems and the improvements of the two modeling methods, namely CNNs and transformers when integrating together.

In section 5.2, we provide the initial results for our novel architecture showing its performance rivaling CNN networks either with random initialization or using pretrained weights also its performance rivals state-of-the-art transformer networks especially when training from scratch. Multiple of the model's variations beat the convolution *UNets* while training from scratch especially in Infarction and Myocardium segmentation opposite to other pure transformer networks such as *Swin UNet* and *UNETR 2D* which didn't perform well on random initializations. Training from pretrained weights

boosted the performance of *Swin UNet* greatly but it is still didn't perform as well as the CNNs or *TransUNet* while *NesT* gained better performance maintaining best results on the Myocardium and Infarction segmentation.

The results are indebted to the well designed local self-attention layers capable of extracting meaningful strong features from small datasets and the block aggregation layer that provided a simple method for sharing the local information extracted per block across the neighbouring blocks. The result was a network has a strong feature extraction capabilities due to self-attention and a strong inductive bias due to the tree structure of the network and the inductive bias from the convolution layer in the aggregation block.

In section 5.3, We study the effects of different data augmentation techniques that proved effective in enhancing the performance by regularizing the training and avoid overfitting which is evident from the boost in performance for both *NesT-B Dense* and *NesT-B*. Both of them are larger networks with higher number of parameters that naturally require more data to train and more prone to overfit. All the augmentation techniques enhanced the performance, For the classical methods both the simple random rotation and Affine transformation augmentation boosted the performance of the models by 3 − 4 dice points, Random rotations performed better than the Affine transformation because random rotations preserve the anatomical structure better than Affine transformation also the reason for this is pointed out by *NesT* original paper (Zhang et al., 2022), as their ablation study showed that the network is highly stable compared to other networks such as *DeiT* and this is reflected in the consistent increase in the performance of the architecture independent to the augmentation techniques used. The *Mix-Up* technique enhanced the performance but with extra complexity compared to the other techniques without showing a satisfying performance corresponding to that cost. Although it achieved its goal of enhancing the performance for the No-Reflow class by extra 10 dice points compared to training without augmentation and 1 − 2 dice points depending on the used model variation but with a negative effect on the other classes.

In section 5.4, We present the effects of the loss functions used. It has been a standard to utilize multiple losses together to enhance the overall performance of the model with each focusing on a certain aspect to optimize. A combination of Cross Entropy and Dice Loss penalise the output of the network on the pixel level and a region level while *Seg-Cls*, our novel loss function is acting as a *False Positive FP* reduction mechanism where it penalizes any segmentation mask that miss-classify the case as malignant instead of healthy by measuring the confidence of the predicted segmentation mask as a class and penalizing the miss-classified pixels specifically.

In section 5.5, we present the experiment of pre-

NesT UNet: Pure Transformer Segmentation Network with an application for Automatic Cardiac Myocardial
Infarction Evaluation

20

training the model using different methods on different datasets. First The *ACDC* dataset is used as for supervised pretraining using a segmentation task where both the encoder and decoder paths are trained on an imaging modality that is similar to *LGE-MRI* and using a task that is highly similar to *EMIDEC* also it has been used for self-supervised training by inpainting the original image from only partial information about the image. In addition, both of the methods are used together achieving higher performance, we indebted this improvement to the advantages of the two methods, for the segmentation the decoder is trained on a similar dataset which serves as a better initialization while training on the *EMIDEC* while the inpainting trains the encoder network to learn the underlying structures (anatomical structures) of the image reconstructing the original image. Similar to *ACDC* we apply inpainting and inpainting with segmentation on the *EMIDEC* dataset. The improvement of performance over the *ACDC* pretraining which is around $2 - 4$ dice points across the board due to the difference in domain, even if the domain is highly similar but the difference in the intensity distribution prevented a great score improvement. On the other hand, pretraining on the *EMIDEC* itself achieved the desired goal of challenging the network to learn the underlying structures of the image with acceptable segmentation performance that petrained the decoder before the fine-tuning solely on the segmentation task.

In section 5.6 we present the results of adopting the novel *NesT* decoder which is similar to the encoder proved capable of learning better than a pure convolution decoder with an performance increase in No-Reflow segmentation by 6 dice points with *EMIDEC* segmentation-inpainting pretraining. Both pretraining experiments testify for the ability of the transformers to learn long-range dependencies and their strong ability to model difficult problems even with the class imbalance on pixels per image level. It is also noting that in all experiments the models produced results which are anatomically correct. The Myocardium is always enveloping the left-ventricle. Both the Infarction and No-Reflow are within the Myocardium, which is very difficult to achieve using CNNs without a complex training process.

The overall performance of the final models is compared to the state-of-the-art and beating most of the leader-board of the *EMIDEC* challenge using only a $2D$ network with limited postprocessing. A summary of the results are shown in table 15, comparing the *NesT UNet* results to the baseline, *EMIDEC* Leader-board, State-of-art, and *NesT* models.

Also the performance is good on the anatomical level. The Basal slices are well segmented and always has high dice score compared to other types especially Apex even with the existence of more anatomical parts such as the Right-ventricle, while on the other hand, Apex shows the lowest performance, which is expected. Still,

	Model	Myo 3D DSC	Infarction 3D DSC	No-Reflow 3D Dice
	UNet	82.83 ± 0.63	54.85 ± 2.58	61.49 ± 6.37
Baseline	TransUNet (Chen et al., 2021d)	83.08 ± 0.45	50.83 ± 6.68	64.05 ± 5.89
	Swin UNet (Cao et al., 2021)	82.42 ± 0.72	57.2 ± 4.71	54.58 ± 10.21
	Zhang	0.879±0.027	0.712±0.268	0.785±0.393
EMIDEC Challenge	Feng et al.	0.836±0.124	0.547±0.340	0.722±0.432
	Yang et al.	0.855±0.027	0.628±0.315	0.610±0.463
STOA	*ICPIU-Net*	95.32	78.3	77.83
	NesT-B[1]	86.29	65.64	66.25
NesT Best Models	NesT-B[2]	85.62	65.77	76.88
	NesT-B[3]	86.45	62.97	73.61
	NesT-B[4]	86.41	66.45	79.13

Table 15: Results Summary, where the best models are reported and compared to the state-of-the-art and the baseline. The *NesT* architectures used in this table are *NesT-B UNet* and they are addition to one another, NesT-B[1] uses data augmentation, NesT-B[2] uses data augmentation and the *Seg-Cls* loss, NesT-B[3] uses the pretraining and finally NesT-B[4] uses the NesT Decoder

it is worth noting that the drop in dice score between the Apex and other parts, especially with Infarction is minimum in the *NesT* architectures.

8. Limitations and Future Work

This architecture is only designed to work in $2D$ and this slicing prevent the network from learning from inter-slice information available for $3D$ segmentation. As a part of the future work, we intend to implement the same architecture in $3D$ and utilize *3D UNet* for refining the output of the $2D$ model, which promises an improvement in the overall result. The Interpretability in this architecture wasn't explored enough, and it will be the focus of future works related to this architecture.

9. Conclusions

In this paper, we explored the application of a novel transformer segmentation network on the difficult *EMIDEC* dataset and achieved results comparable to state-of-art results beating almost every contribution on the challenge's leader-board with only a single $2D$ architecture where the state-of-art require the integration of multiple networks with different sub-tasks. We show that a good mix of convolutions and self-attention blocks can yield better results than convolution results with small data. In addition to the architecture's ability to achieve good results from random initialization training, the architecture can still scale up in performance with the amount of data used for training and pretraining techniques similar to other transformer networks.

References

Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Landman, B.A., Litjens, G., Menze, B., Ronneberger, O., Summers, R.M., van Ginneken, B., et al., 2021. The medical segmentation decathlon. arXiv preprint arXiv:2106.05735 .

Arega, T.W., Bricq, S., Meriaudeau, F., 2021. Leveraging uncertainty estimates to improve segmentation performance in cardiac mr, in: Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Perinatal Imaging, Placental and Preterm Image Analysis. Springer, pp. 24–33.

Ba, J.L., Kiros, J.R., Hinton, G.E., 2016. Layer normalization. arXiv preprint arXiv:1607.06450 .

Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V., 2019. Voxelmorph: a learning framework for deformable medical image registration. IEEE transactions on medical imaging 38, 1788–1800.

Barhoumi, Y., Ghulam, R., 2021. Scopeformer: n-cnn-vit hybrid model for intracranial hemorrhage classification. arXiv preprint arXiv:2107.04575 .

Brahim, K., Arega, T.W., Boucher, A., Bricq, S., Sakly, A., Meriaudeau, F., 2022. An improved 3d deep learning-based segmentation of left ventricular myocardial diseases from delayed-enhancement mri with inclusion and classification prior information u-net (icpiu-net). Sensors 22, 2084.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al., 2020. Language models are few-shot learners. Advances in neural information processing systems 33, 1877–1901.

Camarasa, R., Faure, A., Crozier, T., Bos, D., Bruijne, M.d., 2020. Uncertainty-based segmentation of myocardial infarction areas on cardiac mr images, in: International Workshop on Statistical Atlases and Computational Models of the Heart, Springer. pp. 385–391.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M., 2021. Swin-unet: Unet-like pure transformer for medical image segmentation. arXiv preprint arXiv:2105.05537 .

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers, in: European conference on computer vision, Springer. pp. 213–229.

Chang, Y., Menghan, H., Guangtao, Z., Xiao-Ping, Z., 2021. Transclaw u-net: Claw u-net with transformers for medical image segmentation. arXiv preprint arXiv:2107.05188 .

Chen, H., Li, C., Li, X., Wang, G., Hu, W., Li, Y., Liu, W., Sun, C., Yao, Y., Teng, Y., et al., 2021a. Gashis-transformer: A multiscale visual transformer approach for gastric histopathology image classification. arXiv preprint arXiv:2104.14528 .

Chen, J., Du, Y., He, Y., Segars, W.P., Li, Y., Frey, E.C., 2021b. Transmorph: Transformer for unsupervised medical image registration. arXiv preprint arXiv:2111.10480 .

Chen, J., He, Y., Frey, E.C., Li, Y., Du, Y., 2021c. Vit-v-net: Vision transformer for unsupervised volumetric medical image registration. arXiv preprint arXiv:2104.06468 .

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021d. Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 .

Chen, X., Wu, Y., Wang, Z., Liu, S., Li, J., 2021e. Developing real-time streaming transformer transducer for speech recognition on large-scale dataset, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE. pp. 5904–5908.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q.V., Salakhutdinov,

R., 2019. Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860 .

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee. pp. 248–255.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Dong, L., Xu, S., Xu, B., 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE. pp. 5884–5888.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

Eaton-Rosen, Z., Bragman, F., Ourselin, S., Cardoso, M.J., 2018. Improving data augmentation for medical image segmentation .

Fan, Z., Gong, Y., Liu, D., Wei, Z., Wang, S., Jiao, J., Duan, N., Zhang, R., Huang, X., 2021. Mask attention networks: Rethinking and strengthen transformer. arXiv preprint arXiv:2103.13597 .

Gao, X., Qian, Y., Gao, A., 2021a. Covid-vit: Classification of covid-19 from ct chest images based on vision transformer models. arXiv preprint arXiv:2107.01682 .

Gao, Y., Zhou, M., Metaxas, D.N., 2021b. Utnet: a hybrid transformer architecture for medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 61–71.

Gheflati, B., Rivaz, H., 2021. Vision transformer for classification of breast ultrasound images. arXiv preprint arXiv:2110.14731 .

Girum, K.B., Skandarani, Y., Hussain, R., Grayeli, A.B., Créhange, G., Lalande, A., 2020. Automatic myocardial infarction evaluation from delayed-enhancement cardiac mri using deep convolutional networks, in: International Workshop on Statistical Atlases and Computational Models of the Heart, Springer. pp. 378–384.

He, K., Gan, C., Li, Z., Rekik, I., Yin, Z., Ji, W., Gao, Y., Wang, Q., Zhang, J., Shen, D., 2022. Transformers in medical image analysis: A review. arXiv preprint arXiv:2202.12165 .

Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141. doi:10.1109/CVPR.2018.00745.

Iantsen, A., Visvikis, D., Hatt, M., 2020. Squeeze-and-excitation normalization for automated delineation of head and neck primary tumors in combined pet and ct images, in: 3D Head and Neck Tumor Segmentation in PET/CT Challenge, Springer. pp. 37–43.

Ihm, H.R., Lee, J.Y., Choi, B.J., Cheon, S.J., Kim, N.S., 2020. Reformer-tts: Neural speech synthesis with reformer network., in: INTERSPEECH, pp. 2012–2016.

Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H., 2021. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nature methods 18, 203–211.

Jiang, H., Zhang, P., Che, C., Jin, B., 2021. Rdfnet: A fast caries detection method incorporating transformer mechanism. Computational and Mathematical Methods in Medicine 2021.

Khan, S., Naseer, M., Hayat, M., Zamir, S.W., Khan, F.S., Shah, M., 2021. Transformers in vision: A survey. ACM Computing Surveys (CSUR) .

Krizhevsky, A., Hinton, G., et al., 2009. Learning multiple layers of features from tiny images .

Lalande, A., Chen, Z., Pommier, T., Decourselle, T., Qayyum, A., Salomon, M., Ginhac, D., Skandarani, Y., Boucher, A., Brahim, K., et al., 2022. Deep learning methods for automatic evaluation of delayed enhancement-mri. the results of the emidec challenge. Medical Image Analysis , 102428.

Li, X., Yan, H., Qiu, X., Huang, X., . Flat: Chinese ner using flat-lattice transformer. arxiv 2020. arXiv preprint arXiv:2004.11795 .

Li, Y., Cai, W., Gao, Y., Hu, X., 2021. More than encoder: Introducing transformer decoder to upsample. arXiv preprint arXiv:2106.10637 .

Lin, T., Wang, Y., Liu, X., Qiu, X., 2021. A survey of transformers. arXiv preprint arXiv:2106.04554 .

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Liu, C., Yin, Q., 2021. Automatic diagnosis of covid-19 using a tailored transformer-like network, in: Journal of Physics: Conference Series, IOP Publishing. p. 012175.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.

Loshchilov, I., Hutter, F., 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 .

Ma, J., 2021. Cutting-edge 3d medical image segmentation methods in 2020: Are happy families all alike? arXiv preprint arXiv:2101.00232 .

Mehta, S., Ghazvininejad, M., Iyer, S., Zettlemoyer, L., Hajishirzi, H., 2020. Delight: Very deep and light-weight transformer .

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., 2018. Improving language understanding by generative pre-training .

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al., 2019. Language models are unsupervised multitask learners. OpenAI blog 1, 9.

Rae, J.W., Potapenko, A., Jayakumar, S.M., Lillicrap, T.P., 2019. Compressive transformers for long-range sequence modelling. arXiv preprint arXiv:1911.05507 .

Roy, A.G., Navab, N., Wachinger, C., 2018. Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks. IEEE transactions on medical imaging 38, 540–549.

Shen, Z., Fu, R., Lin, C., Zheng, S., 2021. Cotr: Convolution in transformer network for end to end polyp detection, in: 2021 7th International Conference on Computer and Communications (ICCC), IEEE. pp. 1757–1761.

So, D., Le, Q., Liang, C., 2019. The evolved transformer, in: International Conference on Machine Learning, PMLR. pp. 5877–5886.

Sun, Q., Fang, N., Liu, Z., Zhao, L., Wen, Y., Lin, H., 2021. Hybridctrm: Bridging cnn and transformer for multimodal brain image segmentation. Journal of Healthcare Engineering 2021.

Tang, Y., Yang, D., Li, W., Roth, H., Landman, B., Xu, D., Nath, V., Hatamizadeh, A., 2021. Self-supervised pre-training of swin transformers for 3d medical image analysis. arXiv preprint arXiv:2111.14791 .

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H., 2021. Training data-efficient image transformers & distillation through attention, in: International Conference on Machine Learning, PMLR. pp. 10347–10357.

Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B., Shlens, J., 2021. Scaling local self-attention for parameter efficient visual backbones, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12894–12904.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

Wightman, R., 2019. Pytorch image models. https://github.com/rwightman/pytorch-image-models. doi:10.5281/zenodo.4414861.

Xie, Y., Zhang, J., Shen, C., Xia, Y., 2021. Cotr: Efficiently bridging cnn and transformer for 3d medical image segmentation, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 171–180.

Xu, G., Wu, X., Zhang, X., He, X., 2021. Levit-unet: Make faster encoders with transformer for medical image segmentation. arXiv preprint arXiv:2107.08623 .

Yakubovskiy, P., 2020. Segmentation models pytorch. https://github.com/qubvel/segmentation_models.pytorch.

Yan, H., Deng, B., Li, X., Qiu, X., 2019. Tener: adapting transformer encoder for named entity recognition. arXiv preprint arXiv:1911.04474 .

Yang, S., Wang, X., 2020. A hybrid network for automatic myocardial infarction segmentation in delayed enhancement-mri, in: International Workshop on Statistical Atlases and Computational Models of the Heart, Springer. pp. 351–358.

Yuan, L., Hou, Q., Jiang, Z., Feng, J., Yan, S., 2021. Volo: Vision outlooker for visual recognition. arXiv preprint arXiv:2106.13112 .

Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D., 2017. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 .

Zhang, Y., 2020. Cascaded convolutional neural network for automatic myocardial infarction segmentation from delayed-enhancement cardiac mri, in: International Workshop on Statistical Atlases and Computational Models of the Heart, Springer. pp. 328–333.

Zhang, Y., Liu, H., Hu, Q., 2021a. Transfuse: Fusing transformers and cnns for medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 14–24.

Zhang, Y., Pei, Y., Zha, H., 2021b. Learning dual transformer network for diffeomorphic registration, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 129–138.

Zhang, Z., Zhang, H., Zhao, L., Chen, T., Arik, S.O., Pfister, T., 2022. Nested hierarchical transformer: Towards accurate, data-efficient and interpretable visual understanding, in: AAAI Conference on Artificial Intelligence (AAAI).

Zheng, Y., Li, X., Xie, F., Lu, L., 2020. Improving end-to-end speech synthesis with local recurrent neural network enhanced transformer, in: ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6734–6738. doi:10.1109/ICASSP40776.2020.9054148.

Zhou, H.Y., Guo, J., Zhang, Y., Yu, L., Wang, L., Yu, Y., 2021. nnformer: Interleaved transformer for volumetric segmentation. arXiv preprint arXiv:2109.03201 .

Zhou, L., Liu, H., Bae, J., He, J., Samaras, D., Prasanna, P., 2022. Self pre-training with masked autoencoders for medical image analysis. arXiv preprint arXiv:2203.05573 .
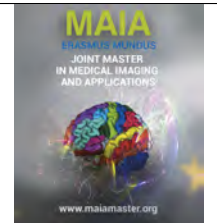
Domain Generalization for Prostate Cancer Detection in MRI

Sheikh Adilina, Anindo Shaha, Henkjan Huisman

Diagnostic Image Analysis Group, Radboud University Medical Center, The Netherlands

Abstract

Modern computer-aided detection/diagnosis (CAD) based on deep learning algorithms achieve high results in detection of prostate cancer in magnetic resonance imaging (MRI). However, the performance of these algorithms drop when the testing cases are taken from a different domain (i.e. samples acquired using a different MRI scanner). In this research, we have investigated the performances of the state-of-the-art domain generalization techniques beginning from the simple solutions like histogram matching to the more advanced deep learning based models like CycleGAN. We do not introduce any new novel method in this study rather we have reapplied the current state-of-the-art techniques and compared the performances. From our experimental results, we have deduced that simple solutions are not adequate to capture the complexity of medical images and hence fail to obtain domain generalization. We have to rely on advanced techniques that take into account not just the intensity information but also the spatial information to achieve our goal.

Keywords: Domain generalization, prostate cancer, magnetic resonance imaging, histogram matching, data augmentation, computer-aided detection and diagnosis

1. Introduction

Prostate cancer (PCa) is the world's second most prevalent cancer and is still one of the leading causes of cancer related death in men (Miller et al.). PCa lesions can range from low-grade, benign tumours that remain harmless forever to highly aggressive tumours that can rapidly advance into clinically significant disease and cause death (joh, 2014). Prostate biopsies are commonly used in clinical practice to histologically assign a Gleason Score (GS) and Gleason Grade Group (GGG) to each lesion as a marker of cancer aggressiveness (Epstein et al., 2016). For decades, ultrasound biopsies were done in absence of a modality that can identify the existence and location of cancer. Prostate magnetic resonance imaging (MRI) is able to succesfully detect and localize prostate cancer and so it is able to rule out needless biopsies (Verma et al., 2017). The standard guideline for reading and obtaining prostate MRI is the Prostate Imaging Reporting and Data System: Version 2 (PI-RADS v2) (Engels et al., 2020) (Weinreb et al., 2016). The shapes and sizes of clinically significant prostate cancer (csPCa) can be very heterogeneous and majority of the times it resembles the various non-malignant conditions making it very complex and time-consuming to interpret. Hence, the development of accurate CAD systems is vital to aid the radiologist in the process of early detecting of csPCa (Saha et al., 2021c).

In recent years, powerful deep learning based algorithms have been developed which are good enough to rival human performance in detecting csPCa. By training on vast volumes of data, deep learning techniques are gaining favor in many fields of medical image analysis. They have produced outstanding results and have been proven to have the capability to generalize.

However, these deep learning models are highly sensitive to domain shifts which might occur due to several reasons. The variability occurs across the different vendors (Siemens Healthineers, Philips Medical Systems, Canon Medical Systems, Toshiba Medical Systems, etc.) of prostate MRI. Even scans obtained using scanner from the same vendor can drastically differ due to the different protocols followed by different institutions. Different patient cohort also plays a role in the problem of domain shift. The best way of tackling this

domain shift is by training a model on all the different variations of the prostate MRI. To make matter worse, the training data in this field is very scarce with ProstateX being the only publicly available dataset.

In this study we investigate state-of-the-art classical and deep learning based algorithms proposed in recent literature for domain generalization in order to propose a robust reproducible algorithm for csPCa detection. That being said, for this study we only focus on the domain shift caused by the difference in scanners across two different vendors, Siemens Healthineers and Philips Medical Systems.

2. State of the art

Several techniques have been introduced over the past few years to bridge the gap of domain shift. To further promote research in domain generalization, challenges like the Multi-Centre, Multi-Vendor and Multi-Disease Cardiac Segmentation (M&Ms) (Campello et al., 2021) and Cross-Modality Domain Adaptation (CrossMoDA) (Dorent et al., 2022) have also been organized where the participants submit their models and are ranked based on the performance on publicly available dataset with hidden test set. The main focus of the M&Ms challenge was to discover deep learning technique which would have the ability to generalize across diverse dataset for cardiac image segmentation while the crossMoDA focused on domain adaptation of brain structure segmentation. From the current literature, two most commonly used approaches of domain generalization are:

1. Test Time Augmentation (TsTA) - where the unknown domain sample is transformed to match the style of known domain during inference.
2. Train Time Augmentation (TrTA) - where the style of the unknown domain is introduced in the model by means of augmentation during training.

The TsTA is a simple and fast yet effective choice to obtain domain generalization since it does not require a model to be trained again from scratch and also does not require a huge amount of data from the target or unknown domain. The top 3 participants of the M&Ms challenge used TsTA, where they generate multiple transformation of a single sample during prediction and then averaged all the prediction results to obtain the final result. A very common method to transfer style during inference is histogram matching (Full et al., 2020) (Meyer et al., 2021) (Ma, 2020).

In the TrTA, the variance is introduced inside the model through augmentation during the training phase. Strategies like histogram matching, blurring, brightness modification, flipping, scaling, rotating, etc are used to introduce diversity in the training data and to enhance the generalization capability of the model (Yaras et al., 2021). Other simple domain generalization techniques

include using label propagation to introduce new samples during training (Zhang et al., 2020). Advanced architectures like, CycleGAN and variational autoencoders are also used for TrTA.

While, the advanced technique were not common among the top performers of M&Ms challenge, majority of the participants of the CrossMoDA challenge used the advanced architectures for domain adaptation. Choi (2021) used the contrastive learning for unpaired image-to-image translation (CUT) (Park et al., 2020) model which is a generative adversarial network (GAN) (Goodfellow et al., 2014) based technique where the network uses contrastive learning to learn the mapping of one domain to another. Wu et al. (2021) and Xu et al. (2021) also used CUT for the translation of the images. Shin et al. (2021), Dong et al. (2021), Liu et al. (2021), Belkov et al., Joshi et al. (2021), Li et al. (2021a) and Ly et al. (2021) used CycleGAN based architecture to perform the translations across domains. In Ouyang et al. (2021), additional shallow networks are used so that the model is able to learn both domain dependent and domain independent features in the samples.

There are other research works like the AutoAugment proposed in by Cubuk et al. (2018), where the best augmentation policies are learned based on the dataset provided. Learning the parameters of optimizing the magnitude and probability of applying each augmentation is extremely time consuming hence there are some variation which are less exhausting like Fast-AutoAugment (Lim et al., 2019). Zhang et al. (2019) used deep stacked transformations where several augmentations are applied on top of each other to achieve more complex augmentations.

Having said that, there are very few literature which focusing specifically on PCa in MRI. Girometti (2020) combined data augmentation with transfer learning to improve domain generalization performance of the model. Chiou et al. (2020) uses CycleGAN to obtain domain adaptation of prostate lesion detection. Hao et al. (2020) explores the standard data augmentation techniques and how these augmentations can be applied independently on the different channels of the MRI to enchance performance. In Grebenisan et al. (2021), instead of optimising data augmentation parameters the authors a sepaarte encoder decoder netowrk is intorduced to make the base model concentrate on structural aspects that remain unchanged even when the domain changes.

In this research, we investigate the impact of common classical techniques (e.g. histogram matching) in domain generalization and the effect of the deep learning based techniques (e.g. CycleGANs) in domain adaptation through data augmentation.

3. Material and methods

3.1. Dataset

The dataset consisted of 3050 prostate multiparametric MRI (mpMRI) scans from Radboud University Medical Center (RUMC) and 988 scans from University Medical Center Groningen (UMCG). The RUMC dataset was used as the primary domain on which all the models were trained and the UMCG dataset was our unknown domain which was used to evaluate the performance of domain generalization capability of the models. Among the 988 UMCG scans, 221 scans were used as the validation set and the rest of the scans were kept aside hidden as the test set.

The RUMC scans were acquired using 3T MR scanners (Skyra 3T, TrioTim 3T and Prisma 3T) of Siemens Healthineers while the the scans from UMCG were acquired using scanners from two different vendors, Siemens Healthineers and Philips Medical Systems. Table 1 shows more detail on the type of scanners present in the UMCG dataset. All RUMC scans were fully annotated by expert radiologists and the UMCG cases were annotated by pathologists. The patients are biopsy-naivemen (RUMC: {median age: 66 yrs, IQR: 61-70} and UMCG: {median age: 69 yrs, IQR: 65-74}), with elevated levels of PSA (RUMC: {median level: 8 ng/mL, IQR: 5-11} and UMCG: {median level: 9.1 ng/mL, IQR: 6-15}).

All the scans were obtained following standard mpMRI protocols in compliance with PI-RADS v2 (Engels et al., 2020). To comply with the most recent literatures in PCa, we have used biparametric MRI (bpMRI) (Saha et al., 2021c) (Saha et al., 2021a) Eklund et al. (2021) which means our dataset includes T2-weighted (T2W), diffusion-weighted images (DWI) and apparent diffusion coefficient (ADC) maps (Jambor, 2017) (Israël et al., 2020). DWI images are acquired at different b values ranging from 50 to 1000 s/mm^2 and T2W images are acquired using a wide range of turbo spin-echo sequence. ADC maps and high b-value DWI (b$\geq$ 1000 s/mm^2) are computed from the raw DWI scans. Prior to usage, all scans are spatially resampled to a common axial in-plane resolution of 0.5 mm^2 and slice thickness of 3.6 mm via B-spline interpolation.

3.2. Model Architecture

To fulfill the main goal of this research work we experimented with mainly two different techniques to transfer the style of the source domain to the target domain. We started out with a very simple approach, histogram matching and then moved to the more advanced generative adversarial networks. We further explored the effectiveness of the different data augmentations. Throughout all these experiments, we kept our baseline segmentation network constant and explored only the other aspects of the algorithm. In this section, we explain our baseline neural network in detail.

Table 1: MRI scanner and vendor information of the University Medical Center Groningen dataset

Vendor	Scanner name	Number of patients
Philips	Ingenia 3T	591
Philips	Intera 1.5T	25
Philips	Achieva dStream 1.5T	11
Philips	Achieva 1.5T	110
Siemens	Avanto 1.5T	43
Siemens	Skyra 3T	90
Siemens	Aera	39
Siemens	Prisma	59
Siemens	Espree	20

For segmentation, we used nnU-Net as our baseline network (Isensee et al., 2021). The nnU-Net model is a self configuring segmentation model based on the U-Net architecture (Ronneberger et al., 2015) (Full et al., 2020). It is able to choose the suitable architecture and augmentation parameters based on the input dataset and hence is independent of any sort of dataset related bias. The nnU-Net model outperformed existing solutions in 23 international biomedical segmentation competitions which consisted of broad variety of datasets which further proves the bias-free nature of nnU-Net. Needless to say, it is also the current state-of-the-art in prostate lesion segmentation (Bosma et al., 2021a).

For our experiments, we used the 3D U-Net configuration which trains for 1000 epoch per fold and as a 5-fold cross validation. The nnU-Net uses Dice and Cross-Entropy (CE) loss functions by default, but we used only the CE loss function as it was shown in Saha et al. (2021b) to be a better loss function when it comes to segmentation in prostate MRI. The Dice loss function fails in our case because not all the samples in our dataset contains lesion and the Dice loss function expects every sample to contain a lesion otherwise it malfunctions specially in cases where the predicted image contains a lesion but the ground truth does not.

The nnU-Net has been designed to automatically adapt its parameters based on the dataset provided (Isensee et al., 2019) and thus have generalization capability. The model starts by adjusting the batch size and patch size based on the memory of the GPU. It prioritizes patch size over batch size as it is vital in improving performance. Then it applies mainly 2 different types of augmentation, intensity based and spatial based. The summary of augmentations applied in the nnU-Net pipeline as provided in the Section 4 of supplementary information of (Isensee et al., 2021) are show in Table 2. The $x \sim U(a, b)$ in the table indicates that x was drawn from a uniform distribution between a and b. All the augmentation functions were build using the python framework called batchgenerators development by the authors of nnU-Net (Isensee et al., 2020). Figure
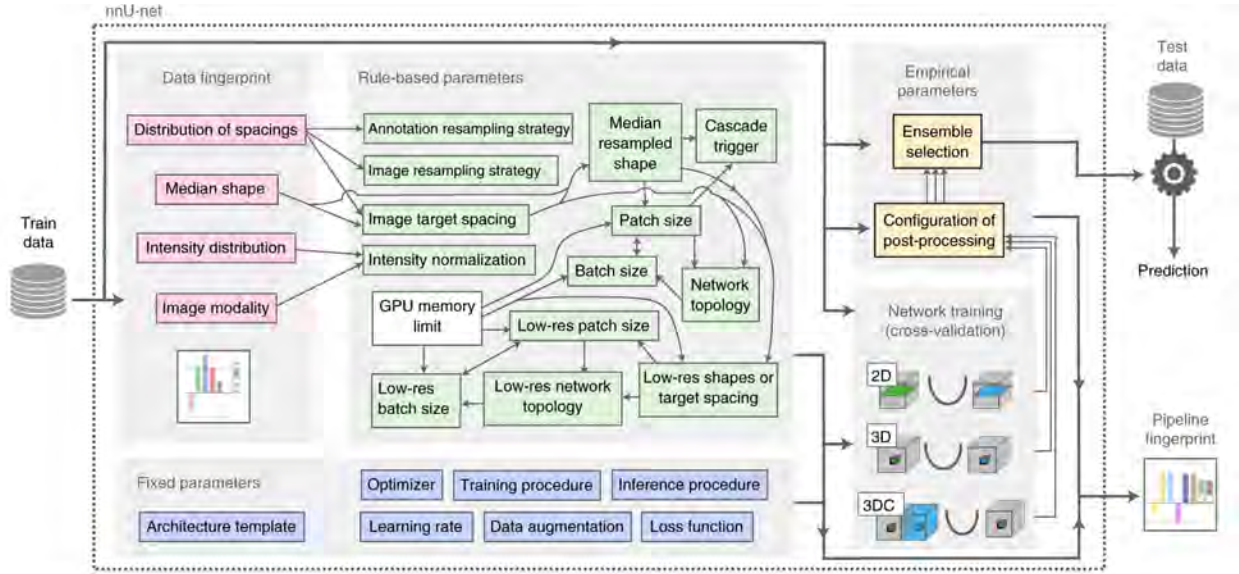
Design choice	Required input	Automated (fixed, rule-based or empirical) configuration derived by distilling expert knowledge (more details in online methods)
Learning rate	–	Poly learning rate schedule (initial, 0.01)
Loss function	–	Dice and cross-entropy
Architecture template	–	Encoder–decoder with skip-connection ('U-Net-like') and instance normalization, leaky ReLU, deep super-vision (topology-adapted in inferred parameters)
Optimizer	–	SGD with Nesterov momentum ($\mu = 0.99$)
Data augmentation	–	Rotations, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirroring
Training procedure	–	1,000 epochs × 250 minibatches, foreground oversampling
Inference procedure	–	Sliding window with half-patch size overlap, Gaussian patch center weighting
Intensity normalization	Modality, intensity distribution	If CT, global dataset percentile clipping & z score with global foreground mean and s.d. Otherwise, z score with per image mean and s.d.
Image resampling strategy	Distribution of spacings	If anisotropic, in-plane with third-order spline, out-of-plane with nearest neighbor Otherwise, third-order spline
Annotation resampling strategy	Distribution of spacings	Convert to one-hot encoding → If anisotropic, in-plane with linear interpolation, out-of-plane with nearest neighbor Otherwise, linear interpolation

Image target spacing	Distribution of spacings	If anisotropic, lowest resolution axis tenth percentile, other axes median. Otherwise, median spacing for each axis. (computed based on spacings found in training cases)
Network topology, patch size, batch size	Median resampled shape, target spacing, GPU memory limit	Initialize the patch size to median image shape and iteratively reduce it while adapting the network topology accordingly until the network can be trained with a batch size of at least 2 given GPU memory constraints. for details see online methods.
Trigger of 3D U-Net cascade	Median resampled image size, patch size	Yes, if patch size of the 3D full resolution U-Net covers less than 12.5% of the median resampled image shape
Configuration of low-resolution 3D U-Net	Low-res target spacing or image shapes, GPU memory limit	Iteratively increase target spacing while reconfiguring patch size, network topology and batch size (as described above) until the configured patch size covers 25% of the median image shape. For details, see online methods.
Configuration of post-processing	Full set of training data and annotations	Treating all foreground classes as one; does all-but-largest-component-suppression increase cross-validation performance? Yes, apply; reiterate for individual classes No, do not apply; reiterate for individual foreground classes
Ensemble selection	Full set of training data and annotations	From 2D U-Net, 3D U-Net or 3D cascade, choose the best model (or combination of two) according to cross-validation performance

Figure 1: Self configuring architecture of nnU-Net

2.4

Table 2: Default augmentations applied in the nnU-Net pipeline

Augmentation	Range	Probability per sample
Rotation	$U(30, 30)$	0.2
Scaling	$U(0.7, 1.4)$	0.2
Gaussian noise	$U(0, 0.1)$	0.15
Gaussian blur	$U(0.5, 1.5)$	0.2
Brightness	$U(0.7, 1.3)$	0.15
Contrast	$U(0.65, 1.5)$	0.15
Simulation of low resolution	$U(1, 2)$	0.25
Gamma augmentation	$U(0.7, 1.5)$	0.15
Mirroring	-	0.5

7b shows the parameters and their automatic configuration in detail.

In terms of our dataset, T2W and DWI scans are subjected to instance-based z-score normalization, whereas ADC maps are subjected to robust global z-score normalization based on the entire training dataset. For anisotropic dataset like ours, nnU-Net applies affine transformations in 2D and other intensity and spatial based augmentations (Bosma et al., 2021b). Also to prevent nnU-Net from zero padding our samples, extended the field of view to 80.0 mm × 80.0 mm × 72.0 mm, which corresponds to a matrix size of 160 × 160 × 20 of our dataset.

3.3. Histogram Matching

Histogram matching (HM) is the simplest way of transferring the style of source domain into the target domain. It is simply done by matching the cumulative histogram of target domain to that of the source domain (Yaras et al., 2021). The built in function in the scikit-image library was used in the implementation of HM (van der Walt et al., 2014) in our experiments. And for all the experiments the HM was done independently for each channel, T2W, ADC and DWI to reduce the complexity of learning. The qualitative assessment that we did and from the scnas shown in Figure 5 and 6, we derived that the translation of T2W and ADC channel across the datasets is fairly easy while the DWI translation is likely the most difficult to learn.

3.3.1. Histogram Matching at Test Time

At this point of the experiment, the nnU-Net model was already trained on the 3050 RUMC cases. Instead of directly predicting on the UMCG samples, we used HM to transfer the style of RUMC cases to the UMCG cases as shown in Figure 2. We began with one-to-one HM. Inspired from the experiments done in Yaras et al. (2021), for each UMCG sample we randomly picked a RUMC case and performed HM. We then performed one-to-one HM by only considered 95% percentile of the reference histogram.

We also did histogram matching of one UMCG case to multiple RUMC cases in hopes to achieve a more

stable outcome. Firstly, we tried the one-to-10 HM approach, where for each UMCG case we randomly picked 10 RUMC cases and got 10 histogram matched samples. We predicted on all the 10 samples and took the average of the softmax predictions. Secondly, we did HM of each UMCG case to the histogram of the entire RUMC dataset. As the last experiment of our TsTA, we matched the global histogram of the UMCG dataset to the global histogram of the RUMC dataset. All the results are discussed in the Section 4.

3.3.2. Histogram Matching at Train Time

Inspired from the approach of the runner-up in the M&Ms challenge (Ma, 2020), we planned to integrate the HM into the already existing powerful augmentation pipeline of nnU-Net. We began by one-to-one HM where we randomly picked a UMCG sample (recall that during TrTA our primary dataset is RUMC) and used HM to introduce the style of the unknown domain during the training phase. To achieve a more generalized result, we experimented with the global histogram of the UMCG dataset instead of using histogram from a single sample. Lastly, we matched the histogram of the whole RUMC dataset to the whole UMCG dataset and qualitatively the matched images looked very stable (examples shown in Figure 4). Therefore we integrated the many-to-many HM method in the final augmentation pipeline . The pipeline is summarized in Figure 3. Having said that, this method caused an issue. We could no longer perform the style transfer during training as we were matching histogram of the whole dataset of source and target domain and thus had to generate the histogram matched images beforehand. All the results are discussed in Section 4.

3.4. Adding More Augmentation

Tweaking the parameters of data augmentations is very reliable when it comes to building a generalized model. Handful of papers have been published where the domain generalization is achieved by solely optimizing the parameters of the augmentations based on the data (Cubuk et al., 2018) (Lim et al., 2019) (Cubuk et al., 2019). The optimization is done based on the
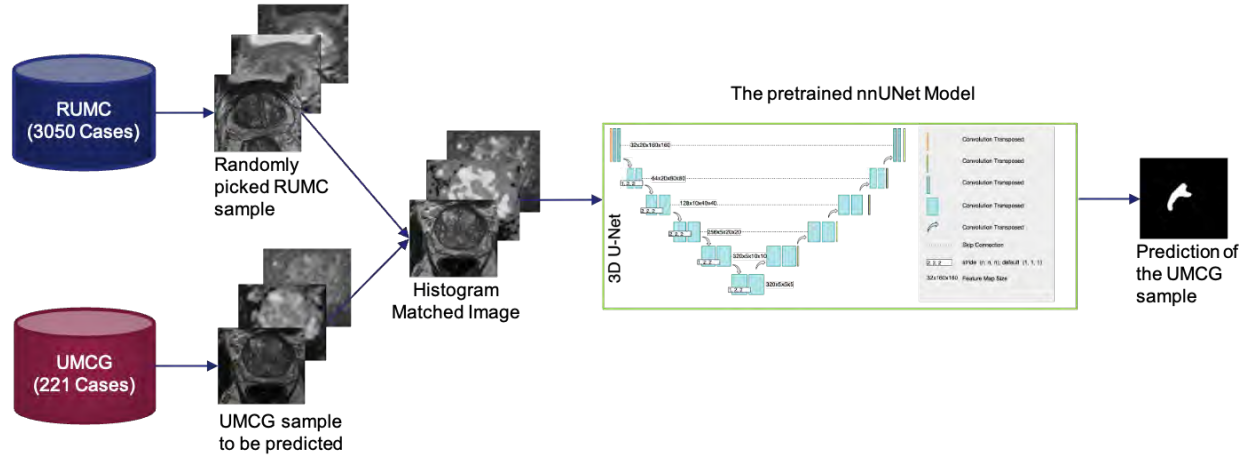
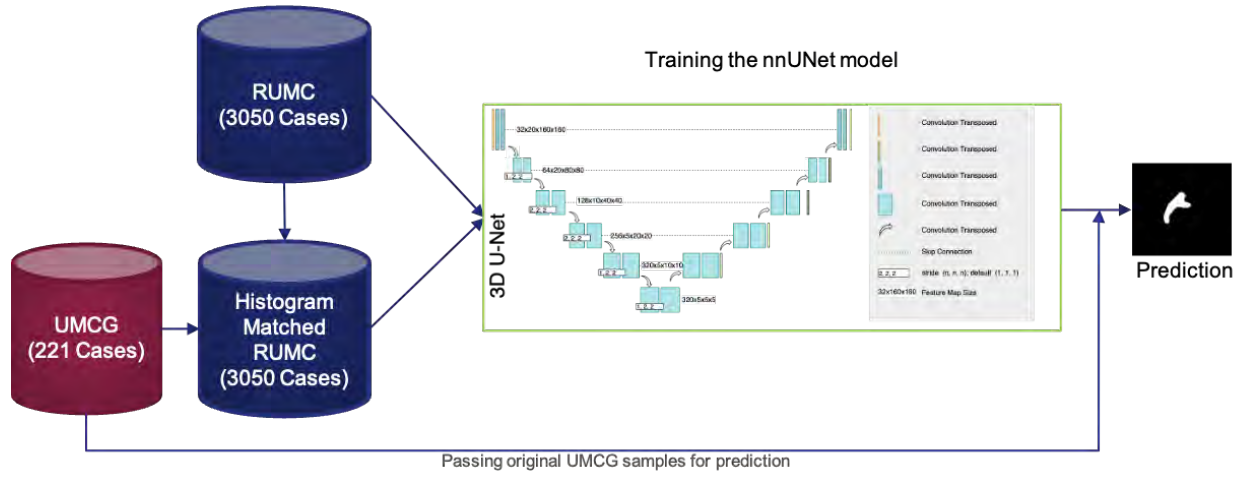Figure 2: Block diagram of histogram matching at test time

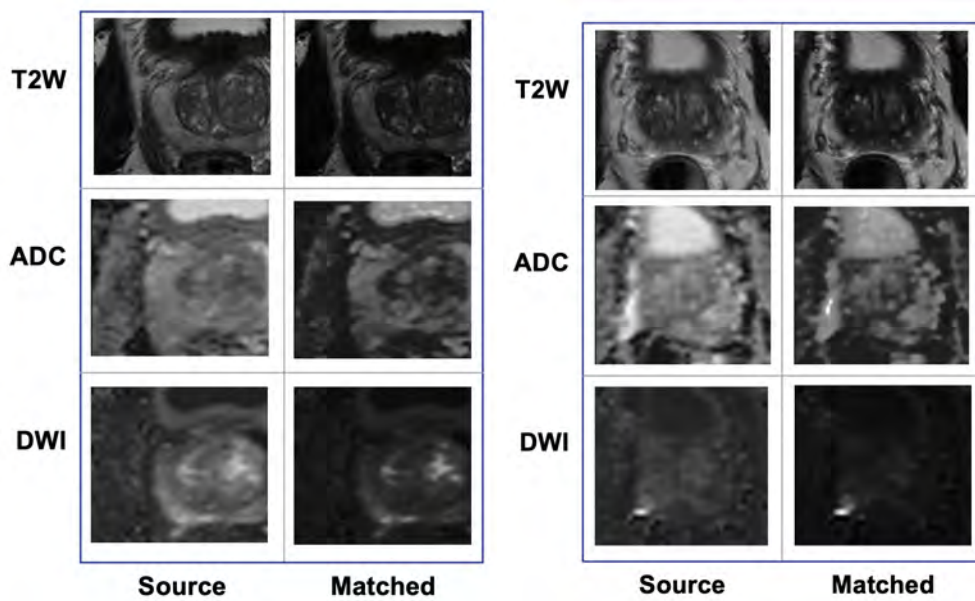Figure 3: Block diagram of histogram matching at train time

Figure 4: Qualitative results of many-to-many histogram matching at train time
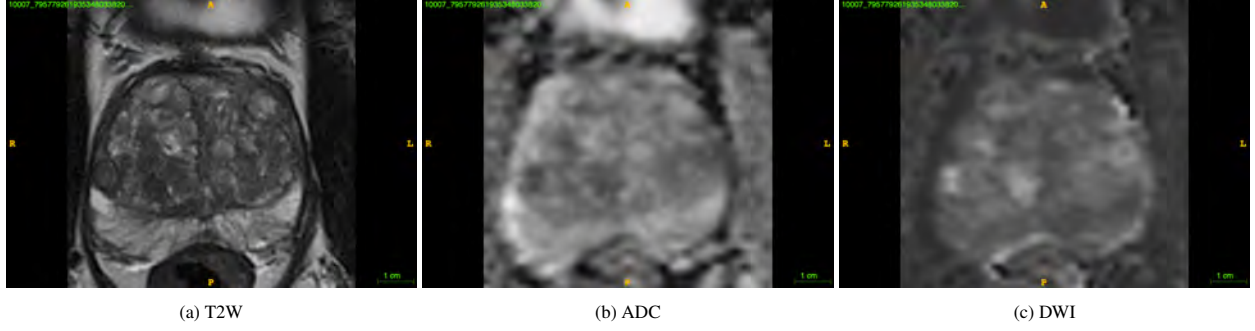
2.6

(a) T2W
(b) ADC
(c) DWI

Figure 5: The prostate bpMRI scans for a single patient from the RUMC dataset is shown above. The 3 sequences shown (5a) T2-weighted imaging (T2W), (5b) apparent diffusion coefficient (ADC) maps and (5c) diffusion-weighted imaging (DWI) were obtained using Skyra 3T scanner of Siemens Healthineers.
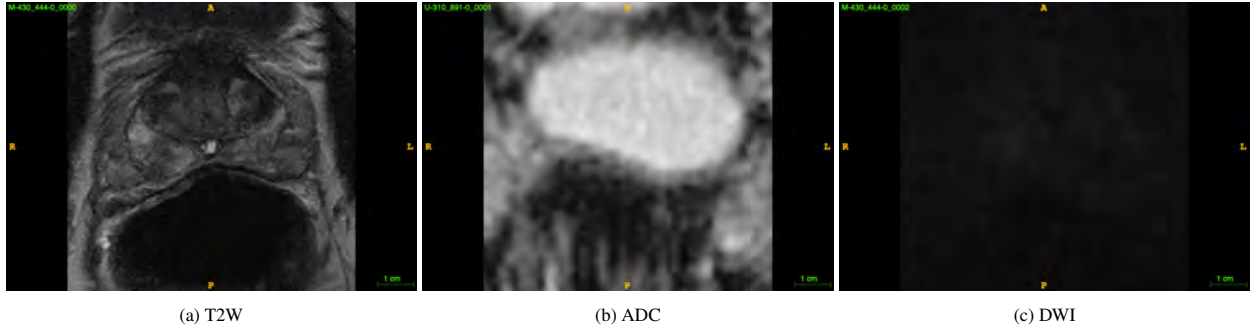

(a) T2W
(b) ADC
(c) DWI

Figure 6: The prostate bpMRI scans for a single patient from the UMCG dataset is shown above. The 3 sequences shown (6a) T2-weighted imaging (T2W), (6b) apparent diffusion coefficient (ADC) maps and (6c) diffusion-weighted imaging (DWI) were obtained using Ingenia 3T scanner of Philips Medical Systems.

magnitude of augmentation to be applied and how often they should be applied. The authors of Xu et al. (2020) integrated the AutoAugment in the data augmentation pipeline of nnU-Net and noticed improvements in the generalization capability of the model. We tried to run the code provided by the authors of the article, but the implementation utilized an earlier version of nnU-Net that is not compatible with the current version. Since the augmentations we were utilizing required us to use the latest version of nnU-Net, we could not successfully run the code. Additionally, all of the papers, based on optimizing of parameters, required the base model to be executed multiple times in order to reach the optimum value and the time constraints of the project would not allow us to complete the experiments in time. Consequently, we focused on the literature which would be possible to implement. Papers by Zhang et al. (2019), Hao et al. (2020) and Girometti (2020) specifies the different ranges of parameters of the augmentations that work well on medical imaging datasets. We tried out the various augmentations in the suggested range of parameters from recent literature as well as the other variations of augmentations provided in the GitHub repository by the authors of nnU-Net. All the results are discussed in Section 4.

3.5. Experimental Design

To ensure a fair comparison with the baseline model, we maintained the preprocessing, tuning and train-validation pipeline for each candidate system in a given experiment (Saha et al., 2021c). For all our experiments, the only part we modified the augmenation part that comes after preprocessing. Patient-level diagnosis performance is evaluated using the Area under Receiver Operating Characteristic (AUROC) metric. Lesion-level detection performance is evaluated using the Average Precision (AP) metric. All the metrics were calculated in three dimensions across entire image volumes.

4. Results

4.1. Histogram Matching at Test Time

For the HM at test time, the baseline nnU-Net was trained on RUMC dataset and predicted on the UMCG dataset. Figure 7a shows the lesion-level average precision (AP) and the patient-level area under the ROC curve (AUROC) of the different HM methods on the validation set of UMCG. The "baseline_nnUNet" refers to the performance of the nnU-Net on the original UMCG samples and the rest of the labels refer to the different HM methods. From the graph we can see that the performance does not improve when one-to-all or all-to-all
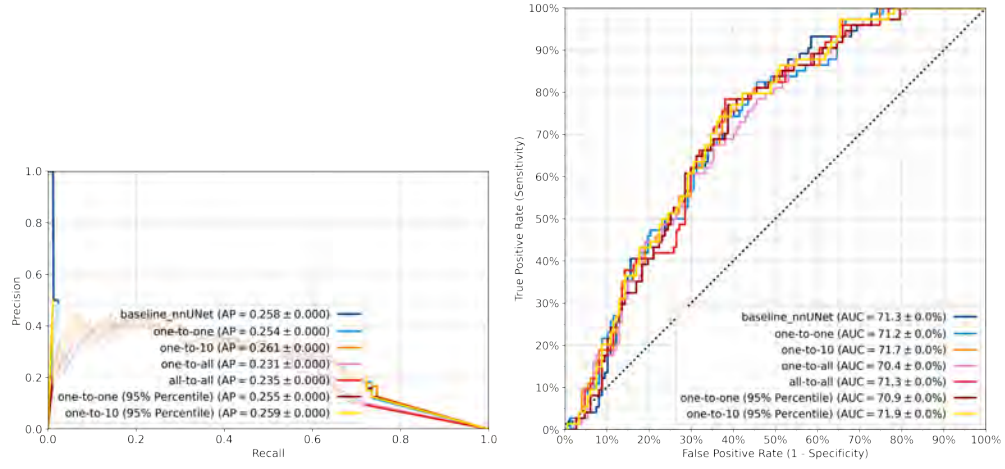
(a) Lesion-level average precision (on the left) and patient-level area under the ROC curve (on the right) of test time augmentation methods
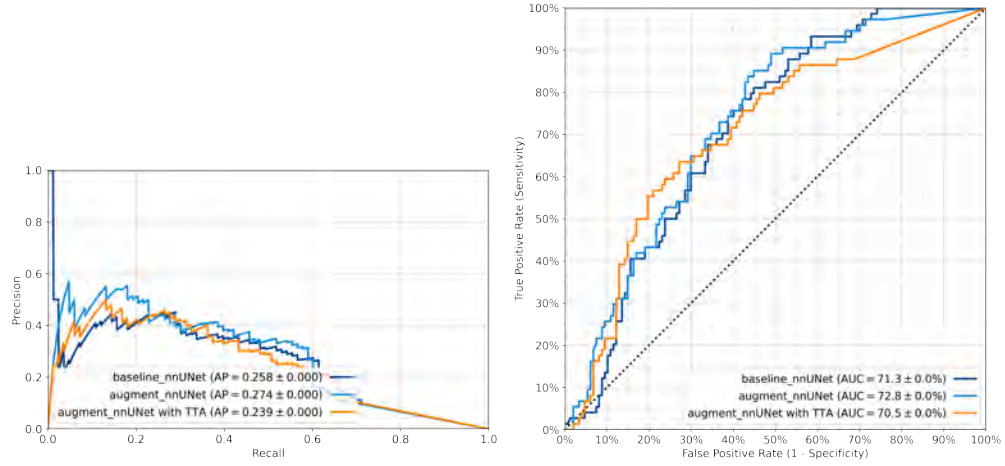
(b) Lesion-level average precision (on the left) and patient-level area under the ROC curve (on the right) of train time augmentation methods
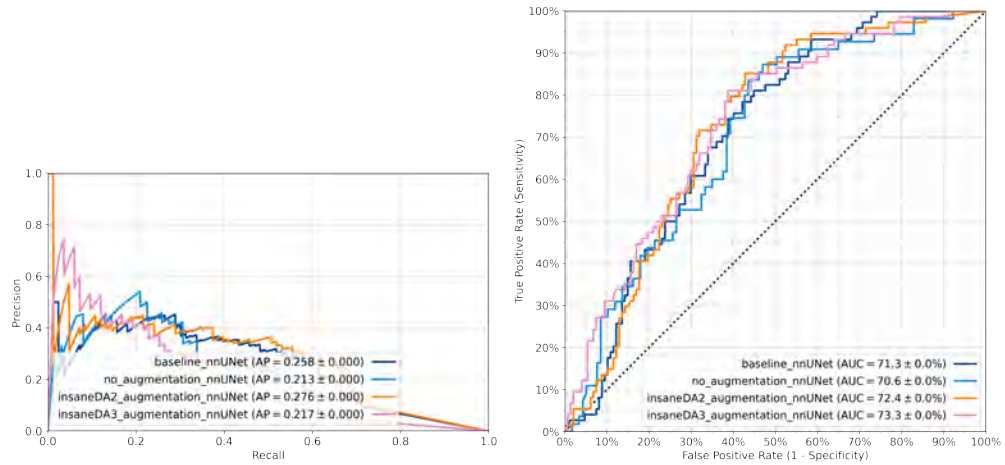
(c) Lesion-level average precision (on the left) and patient-level area under the ROC curve (on the right) of adding more augmentation methods

HM is done. The one-to-one HM is the most unstable one among all the methods as it is completely dependent on the set of samples picked during the matching of histogram. When repeated several times, the performance of one-to-one HM ranges from AUROC of 0.712-0.729 and AP of 0.248-0.268. One-to-10 has the most stable performance and therefore we selected it as our best HM technique for TsTA.

4.2. Histogram Matching at Train Time

For the HM at train time, the "baseline_nnUNet" is as usual the one that was trained on RUMC dataset. The "augment_nnUNet" is the nnU-Net that was trained on the RUMC cases as well as the many-to-many histogram matched RUMC cases, i.e. 6100 cases. Both "baseline_nnUNet" and the "augment_nnUNet" were predicted on the original UMCG cases. The "augment_nnUNet with TTA" is where the "augment_nnUNet" is predicted on histogram matched UMCG samples (one-to-10 in this case). It is clearly visible that the performance drop when we add TsTA. Even though the performance in terms of both AUROC and AP is higher for "augment_nnUNet", the ascent in outcome is not satisfactory.

4.3. Adding More Augmentation

Figure 7c shows the difference between using and not using augmentation in the nnU-Net pipeline. The curve "insaneDA2_nnUNet" refers to the insaneDA2 augmentation that was designed by the authors of nnU-Net. The "insaneDA3_nnUNet" refers to the additional augmentations that we added based on the recent literature. From the graphs we can observe that both of the augmentations have comparable performance.

5. Discussion

We have implemented and investigated as much domain generalization algorithms as possible within the time constraint of the project. In this section, we explain the reasoning behind the performances of the experiments we have done so far.

5.1. Histogram Matching and Data Augmentation

For TsTA, we started out with one-to-one HM which worked fairly well in some cases but failed miserably in cases where the bladder was absent in source image and present in the reference image. Some abnormal bright artifacts also appeared in the resulting image as shown in Figure 8a. After studying the individual histograms (shown in Figure 8), the initial assumption was that the strange peak at the beginning of the reference histogram was the main cause of the artifacts. Considering 95% percentile of the reference image was not able to resolve

this issue as seen in Figure 9a. After studying the histograms of the datasets thoroughly we noticed how different the intensity ranges are (shown in Figure 10a, 10b and 10c). For example, in the case of channel T2W as shown in Figure 10a, the range of intensity is within the range of 0 to 1000 for RUMC dataset whereas it reaches values $\geq$ 2000 in the UMCG dataset. Moreover, the intensity range drastically differ among the individual samples of UMCG which is mostl likely the reason why HM failed to capture the intensity format correctly even in cases were the reference was the global hisotgram of the dataset (examples shown in Figure 9b).

For TrTA, as shown in Figure 11, a high contrast was being introduced when we matched single RUMC case with single UMCG case. When we did the one-to-many HM, we afced yet more issues. First of all, the contrast problem became worse as shown in first image of Figure 12. Secondly, since the nnU-Net contains its own augmentations, in some cases the histogram matching is performed on an already augmented image, resulting in undesirable pixel values present in the source image being taken into account when doing the HM (shown in second image of Figure 12). Due the presence of some vital preprocessing inside the spatial augmentations function this could not be avoided. Moreover, another major reason why HM failed is because it only takes into account the intensity values and cannot handle the challenges of translating medical images which contains distinctive patterns (Isensee et al., 2019).

Neither of the HM techniques was able to considerably boost the performance of our segmentation model on the target domain thus we shifted our focus to the tweaking the parameters of data augmentation. The data augmentations that are proven to work best for domain generalization as mentioned in Girometti (2020), Hao et al. (2020) and Zhang et al. (2019)) are already implemented in the nnU-Net pipeline. So outperforming a model which is already integrated with the best augmentations is difficult. Additionally, there is very few literature available on the topic of prostate MRI segmentation. The very little literature that are present use very weak baseline model to compare their methods. For instance, in the paper by Hao et al. (2020) the performance of the new model is compared to a neural network model with no augmentation. A model trained with augmentation is very likely to perform better than a model with no augmentation. Besides majority of the approaches we found were based on simpler datasets. For example, cardiac images are used in the M&Ms challenge which is much simpler as opposed to our complex bpMRI dataset. As a result, we cannot declare with certainty that the methods described in those articles are well grounded enough to prove that they are actually capable of domain generalization. Even in case where the prostate cancer dataset was used, it was not possible for us to compare our results due to the fact that each author used their own dataset that is not pub-
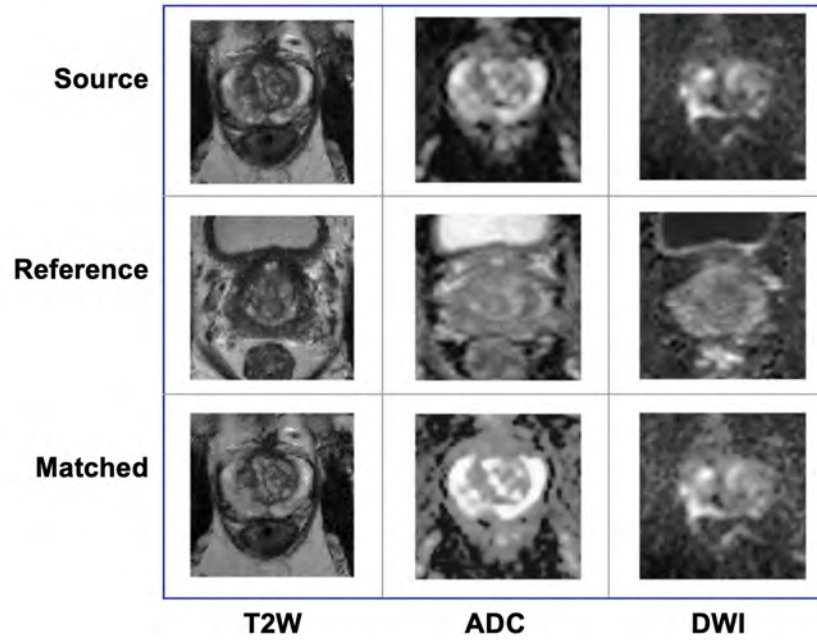
(a) Qualitative results of one-to-one histogram matching
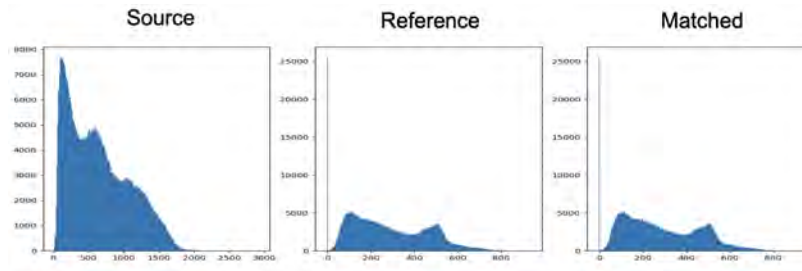
(b) Histogram matching of channel T2W
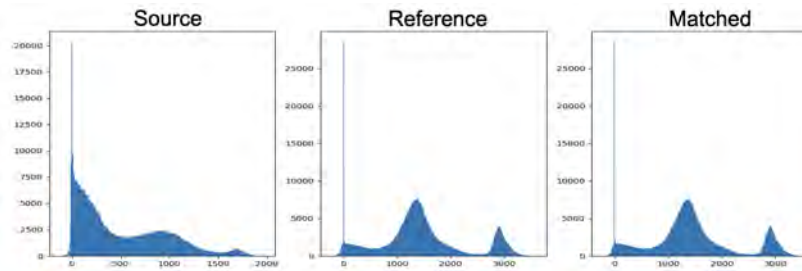
(c) Histogram matching of channel ADC
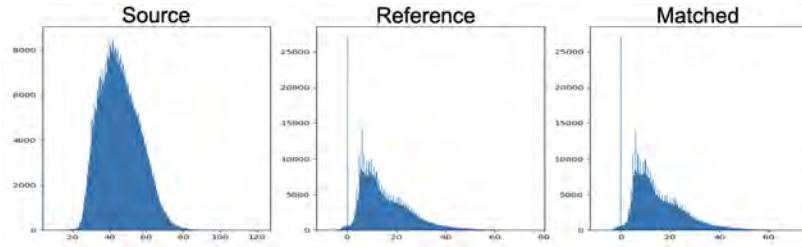
(d) Histogram matching of channel DWI

Figure 8: Qualitative and quantitative results of One-to-one histogram matching in test time augmentation

2.10
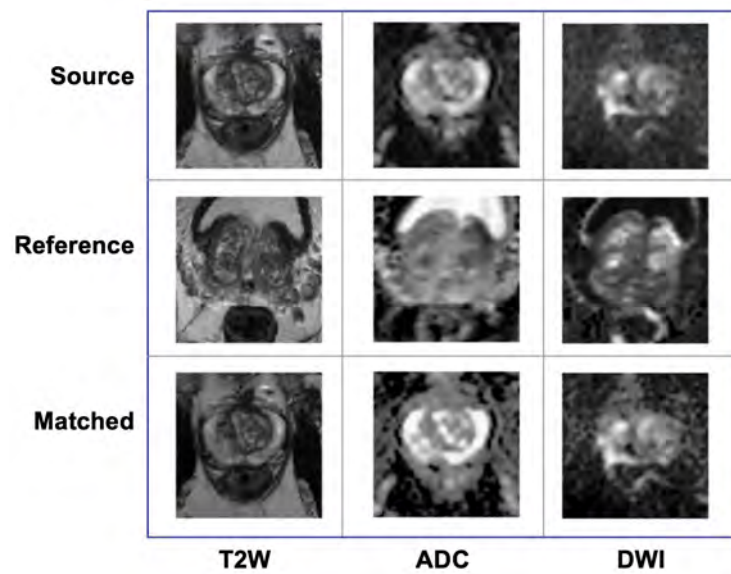
(a) 95% percentile one-to-one histogram matching
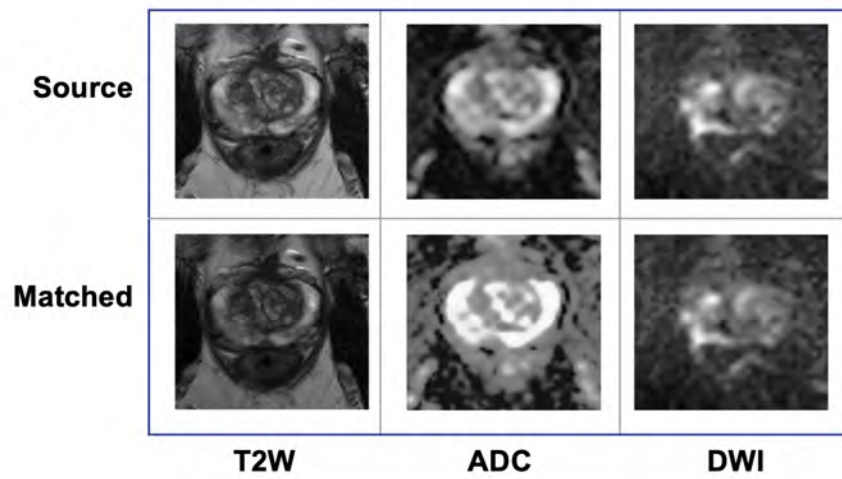
(b) One-to-all histogram matching

Figure 9: Qualitative results of histogram matching at test time

(a) T2W channel of RUMC and UMCG datasets

(b) ADC channel of RUMC and UMCG datasets

(c) DWI channel of RUMC and UMCG datasets

Figure 10: Histograms of T2W, ADC and DWI channels of both RUMC and UMCG datasets
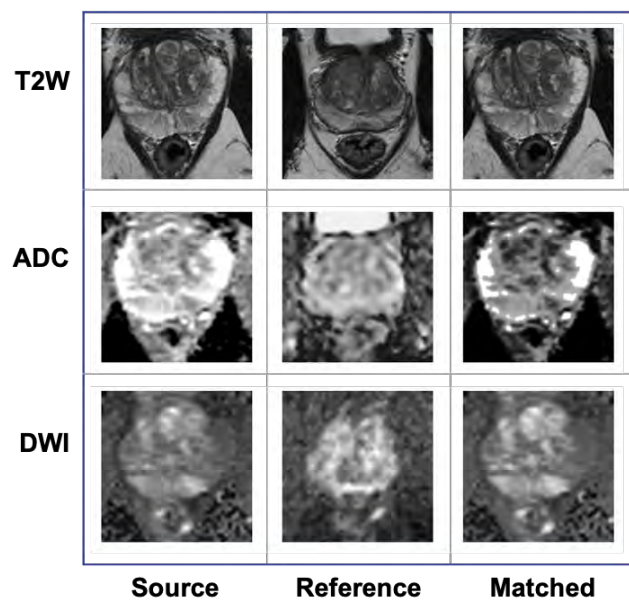
Figure 11: Qualitative results of one-to-one histogram matching at train time
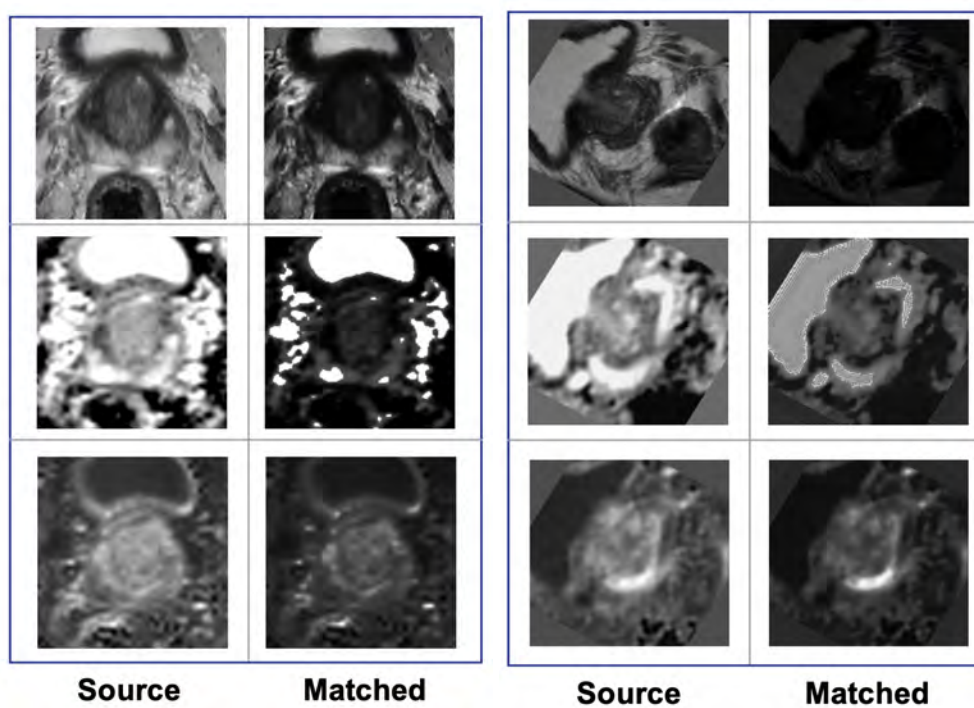
Figure 12: Problems of one-to-many histogram matching during augmentation

2.13

licly available. Given this, having a publicly accessible database for PCa detection in MRI would be beneficial.

5.2. *Limitations and Future Work*

In the long run, the best way of obtaining domain generalization is by training the model on all the different variations of the data. However, in the short run it is not practically feasible due to the lack of data availability. Right now, we must focus on making use of the very little data that is available and still come up with a model capable enough to generalize across domains. From this study, we can clearly conclude that simple approaches like histogram matching is not enough. Hence we must shift to more advanced architecture like Cycle-GAN which is able to capture the intensity as well as the structural transformations across datasets. On a side note, while designing the advanced architecture it is also vital to keep in mind that the transl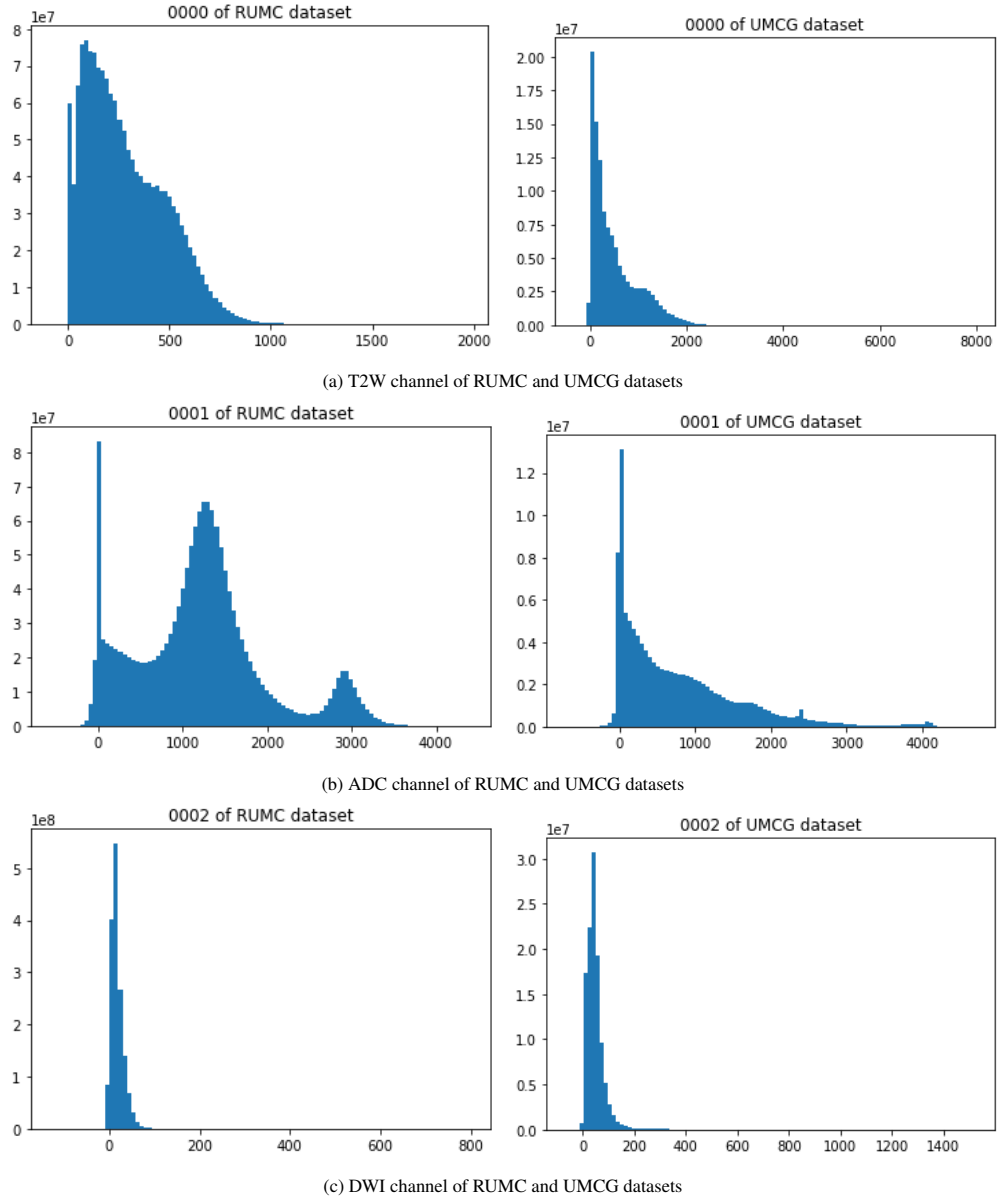ated images does not introduce diagnostically irrelevant or inaccurate information because at the end of the day our ultimate goal is to help improve PCa diagnosis and detection. We have already done several experiments with CycleGAN already as shown in Section Appendix A and we put the rest aside for future study. When we finish building the model, we plan to evaluate it on the publicly available dataset of the recently launched PI-CAI (Prostate Imaging: Cancer AI) grand challenge. In the final evaluation we also plan to add the confidence intervals of the experiments.

One of the major limitations of our project was the lack of time to carry out all the experiments. As a result, we only looked at domain shifts produced by different scanners and on the top of that our dataset was only limited to Siemens and Philips vendors when in reality there are several other vendors. We also did not take into account other external factors (e.g. patient cohort) which are important factors that should be taken into consideration to build a robust domain generalization model for PCa detection.

6. Conclusions

In conclusion, we have reproduced several state-of-the-art techniques in our study. To summarize our experimental results, histogram matching fails to capture the translation required to achieve domain generalization. Moreover, we were unable to outperform the performance of nnU-Net because it already contains the capability to generalize based on the input data. As a result, we have to rely on more advanced architectures if we want to outperform this state-of-the-art self configuring segmentation model.

7. Acknowledgments

References

, 2014. Multiparametric mri in prostate cancer management. Nature Reviews Clinical Oncology 11.

Belkov, A., Shirokikh, B., Belyaev, M., . Comparing unsupervised domain adaptation and style-transfer methods in crossmoda challenge. URL: `https://crossmoda-challenge.ml/media/papers/ira.pdf`.

Bosma, J., Saha, A., Hosseinzadeh, M., Slootweg, I., de Rooij, M., Huisman, H., 2021a. Report-guided automatic lesion annotation for deep learning-based prostate cancer detection in bpmri.

Bosma, J.S., Saha, A., Hosseinzadeh, M., Slootweg, I., de Rooij, M., Huisman, H., 2021b. Annotation-efficient cancer detection with report-guided lesion annotation for deep learning-based prostate cancer detection in bpmri. doi:10.48550/ARXIV.2112.05151.

Campello, V.M., Gkontra, P., Izquierdo, C., Martín-Isla, C., Sojoudi, A., Full, P.M., Maier-Hein, K., Zhang, Y., He, Z., Ma, J., Parreño, M., Albiol, A., Kong, F., Shadden, S.C., Acero, J.C., Sundaresan, V., Saber, M., Elattar, M., Li, H., Menze, B., Khader, F., Haarburger, C., Scannell, C.M., Veta, M., Carscadden, A., Punithakumar, K., Liu, X., Tsaftaris, S.A., Huang, X., Yang, X., Li, L., Zhuang, X., Viladés, D., Descalzo, M.L., Guala, A., Mura, L.L., Friedrich, M.G., Garg, R., Lebel, J., Henriques, F., Karakas, M., Çavuş, E., Petersen, S.E., Escalera, S., Seguí, S., Rodríguez-Palomares, J.F., Lekadir, K., 2021. Multi-centre, multi-vendor and multi-disease cardiac segmentation: The m amp;ms challenge. IEEE Transactions on Medical Imaging 40, 3543–3554. doi:10.1109/TMI.2021.3090082.

Chiou, E., Giganti, F., Punwani, S., Kokkinos, I., Panagiotaki, E., 2020. Harnessing uncertainty in domain adaptation for MRI prostate lesion segmentation. CoRR abs/2010.07411. URL: `https://arxiv.org/abs/2010.07411`, arXiv:2010.07411.

Choi, J.W., 2021. Using out-of-the-box frameworks for unpaired image translation and image segmentation for the crossmoda challenge. URL: `crossmoda-challenge.ml/media/papers/jwc-rad.pdf`.

Cubuk, E.D., Zoph, B., Mané, D., Vasudevan, V., Le, Q.V., 2018. Autoaugment: Learning augmentation policies from data. CoRR abs/1805.09501. URL: `http://arxiv.org/abs/1805.09501`, arXiv:1805.09501.

Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V., 2019. Randaugment: Practical data augmentation with no separate search. CoRR abs/1909.13719. URL: http://arxiv.org/abs/1909.13719, arXiv:1909.13719.

de Bel, T., Bokhorst, J.M., van der Laak, J., Litjens, G., 2021. Residual cyclegan for robust domain transformation of histopathological tissue slides. Medical Image Analysis 70, 102004. doi:https://doi.org/10.1016/j.media.2021.102004.

Dong, H., Yu, F., Zhao, J., Dong, B., Zhang, L., 2021. Unsupervised domain adaptation in semantic segmentation based on pixel alignment and self-training. doi:10.48550/ARXIV.2109.14219.

Dorent, R., Kujawa, A., Ivory, M., Bakas, S., Rieke, N., Joutard, S., Glocker, B., Cardoso, J., Modat, M., Batmanghelich, K., Belkov, A., Calisto, M.B., Choi, J.W., Dawant, B.M., Dong, H., Escalera, S., Fan, Y., Hansen, L., Heinrich, M.P., Joshi, S., Kashtanova, V., Kim, H.G., Kondo, S., Kruse, C.N., Lai-Yuen, S.K., Li, H., Liu, H., Ly, B., Oguz, I., Shin, H., Shirokikh, B., Su, Z., Wang, G., Wu, J., Xu, Y., Yao, K., Zhang, L., Ourselin, S., Shapey, J., Vercauteren, T., 2022. Crossmoda 2021 challenge: Benchmark of cross-modality domain adaptation techniques for vestibular schwannoma and cochlea segmentation. URL: https://arxiv.org/abs/2201.02831, doi:10.48550/ARXIV.2201.02831.

Eklund, M., Jäderling, F., Discacciati, A., Bergman, M., Annerstedt, M., Aly, M., Glaessgen, A., Carlsson, S., Grönberg, H., Nordström, T., 2021. Mri-targeted or standard biopsy in prostate cancer screening. New England Journal of Medicine 385, 908–920. URL: https://doi.org/10.1056/NEJMoa2100852, doi:10.1056/NEJMoa2100852, arXiv:https://doi.org/10.1056/NEJMoa2100852.

Engels, R.R., Israël, B., Padhani, A.R., Barentsz, J.O., 2020. Multiparametric magnetic resonance imaging for the detection of clinically significant prostate cancer: What urologists need to know. part 1: Acquisition. European Urology 77, 457–468. doi:https://doi.org/10.1016/j.eururo.2019.09.021.

Epstein, J.I., Egevad, L., Amin, M.B., Delahunt, B., Srigley, J.R., Humphrey, P.A., 2016. The 2014 international society of urological pathology (isup) consensus conference on gleason grading of prostatic carcinoma: Definition of grading patterns and proposal for a new grading system. The American Journal of Surgical Pathology 40.

Full, P.M., Isensee, F., Jäger, P.F., Maier-Hein, K., 2020. Studying robustness of semantic segmentation under domain shift in cardiac mri. URL: https://arxiv.org/abs/2011.07592, doi:10.48550/ARXIV.2011.07592.

Girometti, R., 2020. Data augmentation and transfer learning to improve generalizability of an automated prostate segmentation model. American Journal of Roentgenology 215. doi:10.2214/AJR.19.22347.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets, in: Advances in Neural Information Processing Systems, Curran Associates, Inc.

Grebenisan, A., Sedghi, A., Izard, J., Siemens, R., Menard, A., Mousavi, P., 2021. Spatial decomposition for robust domain adaptation in prostate cancer detection, in: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pp. 1218–1222. doi:10.1109/ISBI48211.2021.9433779.

Hao, R., Namdar, K., Liu, L., Haider, M.A., Khalvati, F., 2020. A comprehensive study of data augmentation strategies for prostate cancer detection in diffusion-weighted mri using convolutional neural networks. URL: https://arxiv.org/abs/2006.01693, doi:10.48550/ARXIV.2006.01693.

Hu, W., Li, M., Ju, X., . Improved cyclegan for image-to-image translation. URL: weininghu1012.github.io/file/cpsc532L_report.pdf.

Isensee, F., Jaeger, P.F., Kohl, S.A.A., Petersen, J., Maier-Hein, K.H., 2021. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. doi:https://doi.org/10.1038/s41592-020-01008-z.

Isensee, F., Jäger, P., Wasserthal, J., Zimmerer, D., Petersen, J., Kohl, S., Schock, J., Klein, A., Roß, T., Wirkert, S., Neher, P., Dinkelacker, S., Köhler, G., Maier-Hein, K., 2020. batchgenerators - a python framework for data augmentation. URL: https://doi.org/10.5281/zenodo.3632567, doi:10.5281/zenodo.3632567.

Isensee, F., Petersen, J., Kohl, S.A.A., Jäger, P.F., Maier-Hein, K.H., 2019. nnu-net: Breaking the spell on successful medical image segmentation. CoRR abs/1904.08128. URL: http://arxiv.org/abs/1904.08128, arXiv:1904.08128.

Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks, in: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on.

Israël, B., van der Leest, M., Sedelaar, M., Padhani, A.R., Zámecnik, P., Barentsz, J.O., 2020. Multiparametric magnetic resonance imaging for the detection of clinically significant prostate cancer: What urologists need to know. part 2: Interpretation. European Urology 77, 469–480. doi:https://doi.org/10.1016/j.eururo.2019.10.024.

Jambor, I., 2017. Optimization of prostate mri acquisition and post-processing protocol: a pictorial review with access to acquisition protocols. Acta Radiologica Open 6, 2058460117745574. URL: https://doi.org/10.1177/2058460117745574, doi:10.1177/2058460117745574. pMID: 29242748.

Joshi, S., Osuala, R., Martin-Isla, C., Victor M. Campello, C., Sendra-Balcells, Lekadir, K., Escalera, S., 2021. nn-unet training on cyclegan-translated images for cross-modal domain adaptation in biomedical imaging. URL: crossmoda-challenge.ml/media/papers/smriti.pdf.

Kline, T.L., 2021. Improving domain generalization in segmentation models with neural style transfer, in: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pp. 1324–1328. doi:10.1109/ISBI48211.2021.9433968.

Kong, F., Shadden, S., 2021. A Generalizable Deep-Learning Approach for Cardiac Magnetic Resonance Image Segmentation Using Image Augmentation and Attention U-Net. pp. 287–296. doi:10.1007/978-3-030-68107-4_29.

Li, H., Hu, D., Zhu, Q., Larson, K.E., Zhang, H., Oguz, I., 2021a. Unsupervised cross-modality domain adaptation for segmenting vestibular schwannoma and cochlea with data augmentation and model ensemble. doi:10.48550/ARXIV.2109.12169.

Li, H., Zhang, J., Menze, B., 2021b. Generalisable Cardiac Structure Segmentation via Attentional and Stacked Image Adaptation. pp. 297–304. doi:10.1007/978-3-030-68107-4_30.

Li, L., Zimmer, V., Ding, W., Wu, F., Huang, L., Schnabel, J., Zhuang, X., 2020. Random style transfer based domain generalization networks integrating shape and spatial information.

Lim, S., Kim, I., Kim, T., Kim, C., Kim, S., 2019. Fast autoaugment. CoRR abs/1905.00397. URL: http://arxiv.org/abs/1905.00397, arXiv:1905.00397.

Liu, H., Fan, Y., Cui, C., Su, D., McNeil, A., Dawant, B.M., 2021. Cross-modality domain adaptation for vestibular schwannoma and cochlea segmentation. URL: https://arxiv.org/abs/2109.06274, doi:10.48550/ARXIV.2109.06274.

Ly, B., Kashtanova, V., Yang, Y., Aurélien Maillot, M.N.n.G., Sermesant, M., 2021. Cross-modality domain adaptation for vestibular schwannoma and cochlea segmentation from high-resolution t2 mri (epione-liryc team). URL: crossmoda-challenge.ml/media/papers/Epione-Liryc.pdf.

Ma, J., 2020. Histogram matching augmentation for domain adaptation with application to multi-centre, multi-vendor and multi-disease cardiac image segmentation. URL: https://arxiv.org/abs/2012.13871, doi:10.48550/ARXIV.2012.13871.

Meyer, A., Mehrtash, A., Rak, M., Bashkanov, O., Langbein, B., Ziaei, A., Kibel, A.S., Tempany, C.M., Hansen, C., Tokuda, J., 2021. Domain adaptation for segmentation of critical structures for prostate cancer therapy. doi:https://doi.org/10.1038/s41598-021-90294-4.

Miller, K.D., Nogueira, L., Mariotto, A.B., Rowland, J.H., Yabroff, K.R., Alfano, C.M., Jemal, A., Kramer, J.L., Siegel, R.L., . Cancer treatment and survivorship statistics, 2019. CA: A Cancer Journal for Clinicians 69, 363–385. doi:https://doi.org/10.3322/

caac.21565.

Ouyang, C., Chen, C., Li, S., Li, Z., Qin, C., Bai, W., Rueckert, D., 2021. Causality-inspired single-source domain generalization for medical image segmentation. URL: https://arxiv.org/abs/2111.12525, doi:10.48550/ARXIV.2111.12525.

Park, T., Efros, A.A., Zhang, R., Zhu, J.Y., 2020. Contrastive learning for unpaired image-to-image translation, in: European Conference on Computer Vision.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. URL: https://arxiv.org/abs/1505.04597, doi:10.48550/ARXIV.1505.04597.

Saha, A., Bosma, J., Linmans, J., Hosseinzadeh, M., Huisman, H., 2021a. Anatomical and diagnostic bayesian segmentation in prostate mri −should different clinical objectives mandate different loss functions? URL: https://arxiv.org/abs/2110.12889, doi:10.48550/ARXIV.2110.12889.

Saha, A., Bosma, J., Linmans, J., Hosseinzadeh, M., Huisman, H., 2021b. Anatomical and diagnostic bayesian segmentation in prostate mri −should different clinical objectives mandate different loss functions? URL: https://arxiv.org/abs/2110.12889, doi:10.48550/ARXIV.2110.12889.

Saha, A., Hosseinzadeh, M., Huisman, H., 2021c. End-to-end prostate cancer detection in bpmri via 3d cnns: Effects of attention mechanisms, clinical priori and decoupled false positive reduction. Medical Image Analysis 73, 102155. doi:https://doi.org/10.1016/j.media.2021.102155.

Shin, H., Kim, H.G., Kim, S., Jun, Y., Eo, T., Hwang, D., 2021. Self-training based unsupervised cross-modality domain adaptation for vestibular schwannoma and cochlea segmentation. URL: crossmoda-challenge.ml/media/papers/samoyed.pdf.

Verma, S., Choyke, P.L., Eberhardt, S.C., Oto, A., Tempany, C.M., Turkbey, B., Rosenkrantz, A.B., 2017. The current state of mr imaging–targeted biopsy techniques for detection of prostate cancer. Radiology 285, 343–356. doi:10.1148/radiol.2017161684.

van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., the scikit-image contributors, 2014. scikit-image: image processing in Python. PeerJ 2, e453. URL: https://doi.org/10.7717/peerj.453, doi:10.7717/peerj.453.

Weinreb, J.C., Barentsz, J.O., Choyke, P.L., Cornud, F., Haider, M.A., Macura, K.J., Margolis, D., Schnall, M.D., Shtern, F., Tempany, C.M., Thoeny, H.C., Verma, S., 2016. Pi-rads prostate imaging – reporting and data system: 2015, version 2. European Urology 69, 16–40. doi:https://doi.org/10.1016/j.eururo.2015.08.052.

Wu, J., Gu, R., Zhai, S., Lei, W., Wang, G., 2021. A gans-based modality fusion and data augmentation for crossmoda challenge. URL: crossmoda-challenge.ml/media/papers/hi-lib.pdf.

Xu, J., Li, M., Zhu, Z., 2020. Automatic data augmentation for 3d medical image segmentation. URL: https://arxiv.org/abs/2010.11695, doi:10.48550/ARXIV.2010.11695.

Xu, Y., Gong, M., Batmanghelich, K., 2021. Fast single direction translation for brain image domain adaptation. URL: crossmoda-challenge.ml/media/papers/dbmi_pitt.pdf.

Yaras, C., Huang, B., Bradbury, K., Malof, J.M., 2021. Randomized histogram matching: A simple augmentation for unsupervised domain adaptation in overhead imagery. URL: https://arxiv.org/abs/2104.14032, doi:10.48550/ARXIV.2104.14032.

Zhang, L., Wang, X., Yang, D., Sanford, T., Harmon, S., Turkbey, B., Roth, H., Myronenko, A., Xu, D., Xu, Z., 2019. When unseen domain generalization is unnecessary? rethinking data augmentation. URL: https://arxiv.org/abs/1906.03347, doi:10.48550/ARXIV.1906.03347.

Zhang, Y., Yang, J., Hou, F., Liu, Y., Wang, Y., Tian, J., Zhong, C., Zhang, Y., He, Z., 2020. Semi-supervised cardiac image segmentation via label propagation and style transfer. URL: https://arxiv.org/abs/2012.14785, doi:10.48550/ARXIV.2012.14785.

Zhu, J.Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Computer Vision (ICCV), 2017 IEEE International Conference on.

Appendix A. CycleGAN as Augmentation

CycleGAN (Zhu et al., 2017) (Isola et al., 2017) is very commonly used to translate image of one domain to another. Since it has been shown to be very effective for domain generalization, we implemented the algorithm to use it as an augmentation to train our nnU-Net model (Kong and Shadden, 2021) (Li et al., 2021b) (Li et al., 2020). To summarize, CycleGAN consists of 2 generators and 2 discriminator. The first generator is responsible for the translation of source domain to the target domain and the second generator translates the target domain back to the source domain. In addition to the adversarial loss, this neural network has 2 more loss functions, cycle consistency loss and identity loss. The identity loss function makes sure the model does not try to translate images if the input is already given in the target domain. The cycle consistency loss ensures the difference between the initial source image and the resulting source image is minimum.
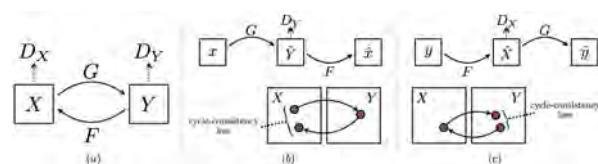
Figure A.13

For our initial experiments, we used 3D CycleGAN and trained separate CycleGANs for each channel. The model failed to learn the translation so we tried other variations of the architecture. The first variation we tried was the residual CycleGAN which was proposed by de Bel et al. (2021). In residual CycleGAN, the input image is considered while generating, by using skip connections, and this resulted in an improvement in overall performance. Having said that, this architecture failed to capture the translation as well. We then experimented using Wasserstein loss function (Hu et al.). We also performed another experiment where we introduce the variations in the training samples by means of random style transfer as showed in Kline (2021). But none of these GAN-based 3-dimensional designs were able to capture the structural properties of the channels. For all the above experiments we relied on the codes provided by the authors in their articles. The different codes from the different authors were coded to handle the specific dataset they were working on and was not designed to handle the massive variations in the intensity levels of our datasets and this could be a major reason why the models failed on our dataset.

This time we took a step back and restarted our experiments by using the code from a much more reliable

(a) Original UMCG   (b) Generated UMCG   (c) Original RUMC
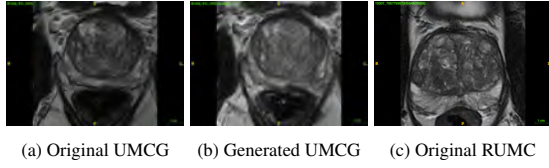
Figure A.14: Generated sample of CycleGAN for the T2W channel

(a) Original UMCG   (b) Generated UMCG   (c) Original RUMC
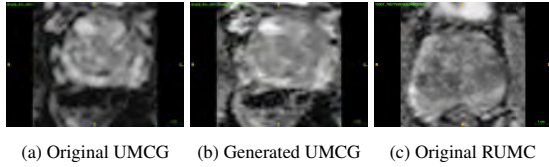
Figure A.15: Generated sample of CycleGAN for the ADC channel

(a) Original UMCG   (b) Generated UMCG   (c) Original RUMC
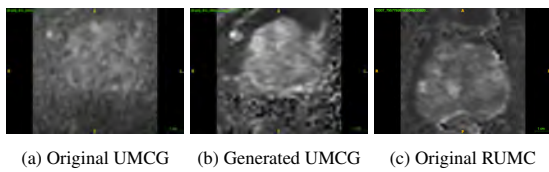
Figure A.16: Generated sample of CycleGAN for the DWI channel
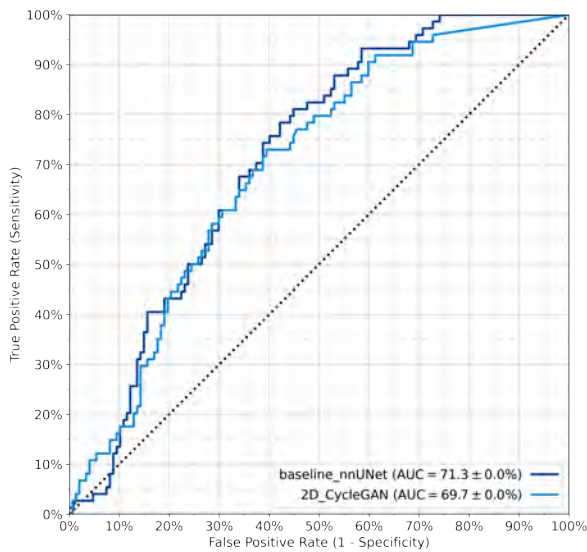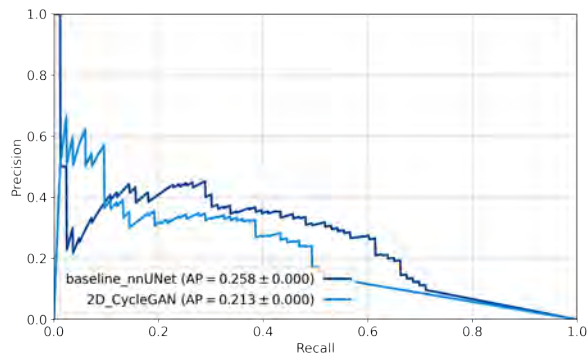
source - the official PyTorch implementation of Cycle-GAN. The library however could only work with 2D PNG images. We picked 50 samples from each dataset and converted each of the 20 slice from a single sample and saved them in PNG. With just 50 samples from each dataset, the CycleGAN model was able to fully capture the transformation and the styles to go back and forth from one dataset to the other. Impressed by the results, we updated the code so that it was now able to handle the NIFTI samples. However, the code was still unable to handle 3D volumes as internally all the transformations and augmentations applied on the data are designed using the Python imaging library (PIL) format which can only handle grayscale or RGB images. In addition to that, the preprocessing of the model required all the samples to be converted, processed and generated in the intensity range of 0 - 255 which means the model never learns the intensity ranges of our original datasets. As a result, the generated images are in an intensity range which our pretrained nnU-Net is unable to recognize. We added a post processing pipeline in the code to retrieve the intensity information. From the generated images in Figure A.14, A.15 and A.16 we can clearly visualize that the model is able to learn the required structural information very well but the results shown in Figures A.18a and A.17a clearly indicates that the intensity information we are loosing in the preprocessing stage is equally important to achieve domain adaptation or generalization and somehow needs to be preserved. We are actively working on this model step by step to create a fully functional three-dimensional CycleGAN that works on our data.

(a) Patient-level area under the ROC curve of CycleGAN compared to the baseline model

(a) Lesion-level average precision of CycleGAN compared to the baseline model

Cell Segmentation and Tracking in Label-free Contrast Images: a Deep Learning Approach

Alexandra Albu, Mario Rosario Guarracino

Università degli Studi di Cassino e del Lazio Meridionale

Abstract

Live cell microscopy is an essential step in analysing the response of cells to certain drugs which lead to advancements in finding a cure for deadly diseases such as cancer. In the present master thesis a deep learning solution is proposed to segment and track cells in microscopy images. The architecture that we propose is based on a DeepSORT tracking which takes as input detections with bounding boxes from a YOLOv4 architecture and indicates the lineage of cells present in a well during their life cycle. The research was developed for DIC cell images provided by the Nikon imaging platform at IBPM-CNR of Rome.

Keywords: cell lines, live cell imaging, mitosis, DIC, segmentation, tracking, YOLO

1. Introduction

Live cell imaging is a key approach in cell biology to study dynamic cellular mechanisms and cell fate in physiological conditions and in processes involving treatment with drug screening for therapeutic aims. For example, according to Gascoigne and Taylor (2008) knowledge of tumor response to certain antimitotic agents would allow better design of clinical trials. These screening tests can only be achieved with live cell imaging techniques. To have a better understanding of the response of human tumor cells to antimitotic agents, a systematic approach is taken. More precisely, by using automated time-lapse microscopy, single-cell-bases assay is established and the response of cells to different antimitotic drugs is analysed. Additionally, it has been tried to identify oncogenic lesions that could be responsible for cell fate outcomes.

Furthermore, quantitative information of cell cycle length, fate of the cells combined with the analysis of different drugs on the cell yet excluding the investment of time and financial resources

Caldon and Burgess (2019) highlight the benefits of lineage analysis and the increased potential of breakthroughs in cures for various diseases. Therefore, computational analysis of live cell imaging is very important.

Mitosis is a fundamental biological process which results in forming two new daughter cells which are replicas of the mother cellSullivan (2001) and is in consequence an important part of live cell imaging analysis.
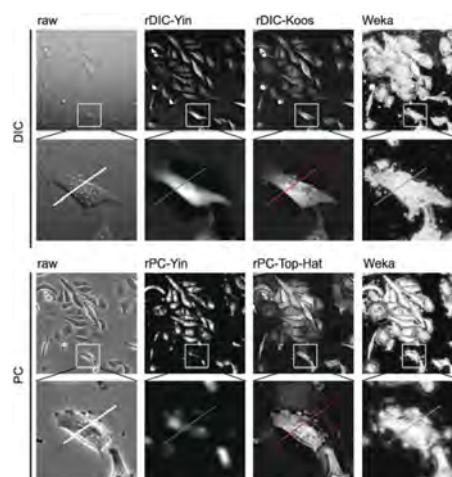

Figure 1: Field of view for raw and reconstructed DIC and PC images

We can observe in Figure 1 how Vicar et al. (2019) show quality of reconstructions of two types of transmitted light microscopy: Differential Interference Contrast (DIC) and Phase Contrast (PC) images. The cells

have a transparent nature which leads to the need of contrast enhancement techniques based on the phase information and unfortunately artifacts are introduced at this stage. As a result, the PC cells have a halo effect and shade-off and the DIC cells have a 3D like topographical effect due to shadow-cast artifacts. Both types of images present significant challenges in terms of segmentation, yet we can observe that the results on DIC images are less desirable in comparison with the PC ones. We can therefore admit that the DIC images present a major challenge in terms of image processing and computer based analysis.

With the importance of the cell lineage and the financial and biological materials in mind, the human factor needs to be duly noted as well. Biologists perform a plethora of tests and although the recognition of the phase of the cell is relatively easy for the human eye, observation and tracking of the same cells present at the beginning of the experiment throughout the whole duration can prove to be a difficult task due to the clarity of the image, crowding of the cells in the image, large number of images or human factors such as tiredness or subjectivity.
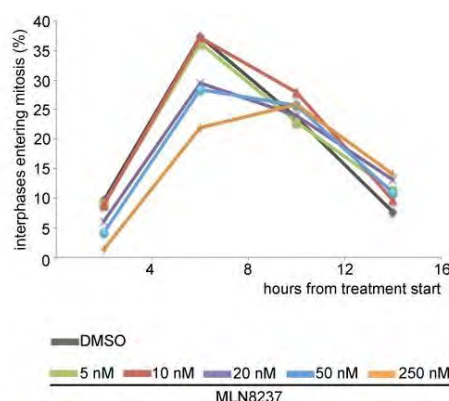

Figure 2: MLN8237 delays entry into mitosis.

Figure 2 was used in order to visualize the effect of different dosages of the the MLN8237 inhibitor and DMSO on various cells to track the percentage of interphasic cells entering mitosis with respect to the time for the experiment. According to Asteriti et al. (2014) the results were grouped in 4-hours intervals and it recorded 250 interphases in 3 experiments for each condition. The magnitude of the experiment is impressive and there are countless other experiments done in this field.

Based on the criteria present in the microbiology research field and understanding the needs of the biologists we propose a Deep Learning solution to detect and track interphasic and mitotic cells and build the lineage of these objects.

To materialize our goal we preprocess and augment the given images, afterwards we feed the prepared dataset to a detector whose output becomes input for the tracker. We then output the tracked cells in video output with the detections overlapping the image as well as CSV format containing information about the lineage. This architecture can be visualised in Figure 3.

In the upcoming sections we present details about the research process and creating this paper while describing the current state-of-the-art, the challenges that our dataset brought, the methods that we have tried, our results, discussion on the results, the conclusions of our work as well as the acknowledgements of the people that made this work possible.

2. State of the art

The **Ilastik** toolkit, described in Berg et al. (2019), is an interactive machine learning software under the GNU General Public License for bio-image processing and analysis. Several experiments were performed trying to polish the capabilities of this tool to datasets under investigation, since it performed very well for other datasets. For example, Baharlou et al. (2019) use ilastik to classify cells in mass cytometry into three classes: nuclear, cytoplasmic and background. Another approach for cytometric images in described in Damond et al. (2019) where with the help of Ilastik, they identify islets and blood vessels. Another example of the usage of this tool is present Stringer et al. (2021) in the context where they utilize Ilastik to segment 3D volumes of a dataset containing stained histological images of human organs, fluorescent U2OS images with Hoechst stain and NIH3T3 fluorescent cells.

Aside from the high performance on various applications, usability was another benefit provided by this tool. Indeed, Ilastik has a user-friendly interface and the possibility to replicate the same steps by sharing *projects*. This is depicted in Figure 18.

Ilastik is using machine learning algorithms such as Random Forest to generate new predictions for segmentation.

The ilastik menu provides several pre-defined workflows to perform image segmentation, classification and tracking.

Object Classification and Tracking of Ilastik require either the results of Pixel Classification workflow, or annotated images also known as segmentation masks. The results of the Pixel Classification workflow may be either segmentation mask or pixel prediction map. Pixel prediction maps assign to each pixel a probability that the pixel is close to the nucleus center, while the segmentation map clusters parts of the image where the pixels belong to the same class.

The images under investigation haven't had any segmentation ground truth. Therefore, we had to create our own annotations.

We have used for the best results the Ilastik pipeline in the following order and we explain in detail in the Appendix the several experiments that we performed.
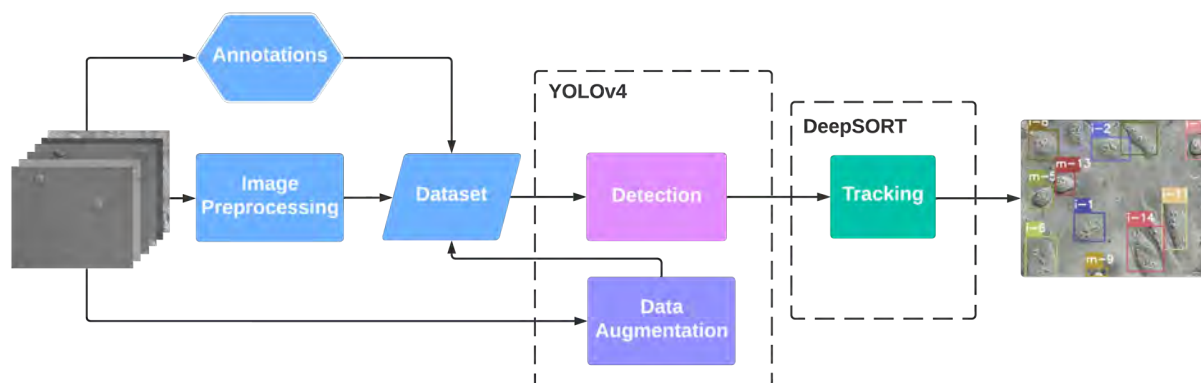
Figure 3: Proposed Deep Learning based architecture

1. Pixel Classification to obtain prediction maps for two classes: cells and background;
2. Object Classification to gather maps for tracking for two classes: interphasic and mitotic;
3. Tracking to output tracking masks and CSV tracking file.

The final tracking for the manually annotated images was fairly good, we see in the following picture where each color represents an assigned cell identity.

Figure 4: Example of generalisation for field E

Unfortunately, Ilastik was not able to generalise well and thus the result by batch processing to obtain new masks based on the training was unsatisfactory as we see in Figure 5 where we show the raw image and the automatic output with prediction map in the middle and segmentation map in the right. This is a major issue because tracking is highly dependent on the proper detection of the objects.
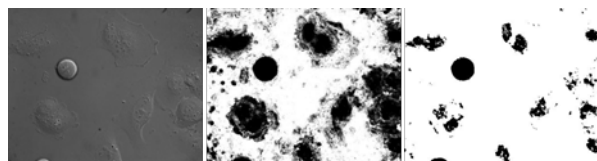
Figure 5: Example of generalisation for field E

Ideally, using ilastik would have been a more stable and easy to use tool for the biologists in the long run, but considering all the previous points we had to continue our search of finding another solution.

A possible approach consisted of using an U-Net model to obtain the segmentation masks. U-Net has shown high segmentation performance in medical imaging, so it was a natural choice to use this algorithm. Be-
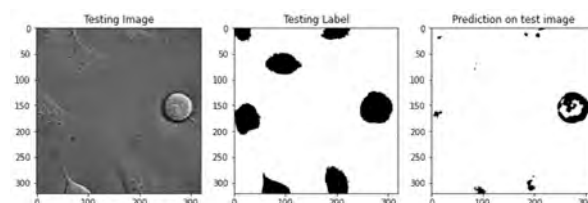
Figure 6: U-Net predictions for binary classification

cause U-Net did not retrieve satisfying results either, we continued our literature review and we had the upcoming findings.

Currently two cell tracking challenges with publicly available datasets exist: Cell Tracking Challenge (CTC) Ulman et al. (2017) and Cell Tracking and Mitosis Challenge (CTMC) Anjum and Gurari (2020). The CTC challenge contains fluroescent microscopy images, while the CTMC provides bright-field microscopy images.

Anjum and Gurari (2020) benchmarked two state-of-the-art cell tracking algorithms that were tested on the challenge dataset: Viterbi and DeepCell where both take as input segmentation masks.

Viterbi Magnusson et al. (2015) is considered to be the best performing algorithm as per the last Cell Tracking Challenge and it uses object associations for tracking.

DeepCell Moen et al. (2019) uses in essence deep multiple object tracker and traditional cell event detector and RetinaMask with pretrained ResNet50 on ImageNet.

Both state-of-the-art algorithms perform well on fluorescent and certain bright field images. However, they

altogether fail to track the U2OS cell line as it has been shown in Anjum and Gurari (2020).

Following the paper Ulman et al. (2017) many traditional approaches and some machine learning based algorithms proved to be efficient for the CTC data. Yet, we had to take into account that the CTC data contains mostly fluorescent images and according to the Ulman et al. (2017) the data that negatively influenced the algorithms and led to worse performance was closer to brightfield images.

Our dataset containing U2OS cells is similar to the CTMC challengee U2OS cell lime images. The previously mentioned challenge together with the CTMC statements lead us to search for other state-of-the-art algorithms outside the cell paradigm.

If we take into consideration only the microscopy imaging and cells segmentation, all algorithms mentioned up until this point in this section would have significant performance on images with a clear distinction between the nucleus and the background. An important limitation to bear in mind is the considerable drop in performance when given data where the background and foreground are very similar in terms of color, illumination, contrast and texture.

The YOLO family of object detection has been the state-of-the-art for multiple of their models in the COCO dataset Lin et al. (2014). Specifically, YOLOX Ge et al. (2021) achieved state-of-the-art performance on the COCO dataset in 2021.

DeepSORT is a high performance tracking algorithm that has a strong mathematical foundation. It was not specifically designed for microscopic cells tracking, but it presented very good results on pedestrian tracking. Wojke et al. (2017)

Considering the literature review, we implement some of the previously mentioned algorithms and we discuss their performance in the upcoming chapters.

3. Material and methods

3.1. Dataset

Our dataset consists of raw Differential Interference Contrast (DIC) images of human osteosarcoma U2OS cell lines seeded in 2-4 micro-slides (Ibitreat) were observed with an inverted microscope (Eclipse Ti, Nikon) using a 40× (Plan Fluor, N.A. 0.60, DIC) or a 60× Oil (Plan Apo, N.A. 1.4, DIC) objective (Nikon). During the whole registration, cells were kept in a microscope incubator (Basic WJ, Okolab) at 37 ∘C in 5% CO2. DIC images were acquired every 5 or 7 min using a DS-Qi1Mc camera (Nikon) or a Clara camera (ANDOR technology). Asynchronous cultures were treated with Aurora kinase inhibitor (MLN8237) to induce mitotic defects and cell death.

Our initial data consisted of TIFF videos that represent cells placed in 5 different fields: A, B, C, D, E and

Field	Frames	Interphasic	Mitotic
A	69	904	98
B	69	997	43
C	69	858	111
D	68	406	198
E	66	506	176
T	67	941	85
Total	408	4612	711

Table 1: Dataset summary

T. The videos add up to 408 frames of size 400×320 with 5323 cumulated objects.

We describe the distribution of images and objects in Table 1 and we can conclude that a high class imbalance is present among the presence of interphasic and mitotic cells.

Figure 7 shows a sample from each of the six fields starting from the red-colored frame with field A and going clockwise to the violet-colored field T. Here, we can observe that the illumination and contrast strongly vary in the different fields.
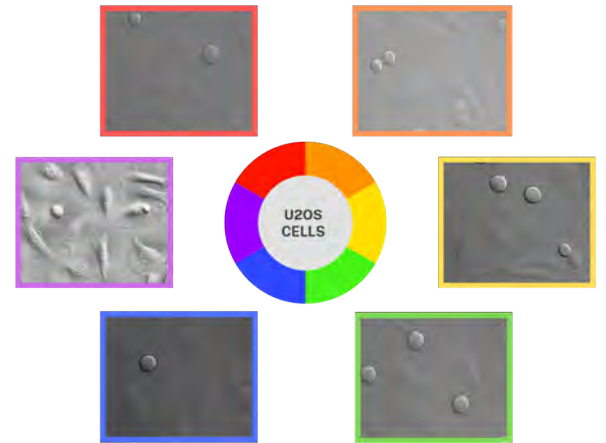
Figure 7: Images of fields A (red), B (orange), C (yellow), D (green), E (blue) and T (violet)

For this project, we were interested in two types of cells: interphasic and mitotic. These are exemplified in Figure 8, where the blue and yellow rectangles indicate an interphasic and a mitotic cell, respectively.

As previously mentioned, the desired output is to detect the duration of mitosis and then the resulting daughter cells. Because of this it was not of high importance the perfect recognition of the cytoplasm. Instead, by convention we could rather take into account the nucleus of the cell as reference.

3.1.1. Annotations

To annotate the data for further use in the soon explained methods in Section 3.3 we have used two types

3.4

Figure 8: Example of the types of cells in field C

of annotation:

1. Segmentation masks
2. Bounding boxes

The segmentation masks provide pixel-wise labeling of the mitotic and interphasic cells for each image. They were hand drawn and refined using the ilastik tool Berg et al. (2019), as detailed in the Appendix.

The bounding boxes provide object-wise labeling of the mitotic and interphasic cells and were drawn and labeled for each image using the Make Sense online tool Skalski (2019). Using this tool, we have drawn all the cells present in each frame and labeled them as either interphasic or mitotic. After this step, we have exported the annotations in YOLO format as text files and as xml files.

Figure 9: Bounding box annotation of a B field image

In Figure 9, we present an example of bounding box annotation of a frame from the B field sequence. Here we can observe the interphasic cells delimited with yellow bounding boxes and the mitotic cells with red bounding boxes. By convention, we label the two daughter cells resulting right after division as mitotic cells as their visual characteristics are very similar to the standard mitotic cell. This does not hamper the quality of the annotation since, after a few frames, they change morphology, and we label them as interphasic cells.

Concerning the creation of ground truths for the tracking, we have taken the annotations obtained from Make Sense Skalski (2019) in the Pascal VOC format and converted them into a CSV file. Afterwards, with a simple script, we have visualised one bounding box at a time for each frame of the test set and manually filled in the tracking id for each cell object. The result of the

tracking annotation of ground truth has the following format:

$< frame >< trackId >< class >< xmin ><$
$ymin >< xmax >< ymax >$

3.1.2. Data Augmentation

We have included data augmentation for three cases: U-Net, YOLOv4, YOLOX. For the U-Net we augmented the data by applying transformations such as random rotation, horizontal and vertical flip, transpose, grid distortion, color space transformations like random contrast, random brightness, random gamma, random crop, sharpen, blur, clahe.

For YOLOv4 data augmentation is present as an important step in the methods of the architecture. They have used both photometric distortion (e.g., brightness, contrast, saturation or noise adjustment) and geometric distortion (e.g., scaling, cropping, flipping and rotating). Furthermore, special methods are used for instance DropBlock applied in the context of feature map for regularization Ghiasi et al. (2018). They also introduce two new data augmentation methods: Mosaic and SAT. Unlike CutMix which cuts patches of the image and mixes only two input images by pasting them among training images while also proportionally mixes the ground truth labels Yun et al. (2019), Mosaic mixes four training images. Self-Adversarial Training (SAT) encapsulates two forward backward stages, where the first one is an image altering neural network and the second is that the network is trained to detect normally the previously modified image.

YOLOX uses mainly MixUp Zhang et al. (2018) and Mosaic, but when comparing the performance of this model with other different previous models, they slightly change a few parameters or even remove completely MixUp.

3.2. Metrics

We differentiate between two categories of metrics: object detection and tracking metrics and in the upcoming paragraphs of this section we explain these metrics. For the first category we are using precision, recall, F1 score and mAP. In order to evaluate the performance of the tracking approach, we use IDF1, MOTA and MOTP.

Precision is the metric which "tells" how many correct predictions a model produced with respect to the total number of predictions. Below we show the explicit formula where TP represent the correct predictions also known as true positives and the FP which is the abbreviation for false positives. In our case a false positive means that a cell is detected when that cell is not in fact present. A perfect precision would have value 1 meaning that there would be no false detections.

$$Precision = \frac{TP}{TP + FP}$$

Recall also known under the name of sensitivity is computed as the ratio between the correct predictions with respect to the total number of cases in which it occurs. In this case the false negatives are denoted as FN and for instance it means when a cell is present in the frame, but it does not get detected. The recall is perfect is when no false negatives are present so the ratio is 1.

$$Recall = \frac{TP}{TP + FN}$$

F1 score The F1 score is computed as the harmonic mean of the precision and recall, where an F1 score is perfect when both the precision and the recall are perfect which means that the value of the metric is 1.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Before we define the mean Average Precision (mAP), we must explain how we regard to the intersection over union. The intersection over union (**IoU**) measures the intersection of the detections of the algorithm compared to the groundtruth and the perfect value is 1. In our case we consider a correct prediction if at least 50% of the detection overlaps with the groundtruth.

$$IoU = \frac{area\_of\_overlap}{area\_of\_union}$$

The Average Precision (AP) can be seen as the Area Under Precision-Recall Curve (AUPRC) and is computed using the following formula:

$$AP = \int_0^1 p(r) \, dr$$

mAP is defined as the mean value of the computed Average Precision for each class present in the dataset. Because the AP can achieve the ideal value of 1 since the precision and recall can also achieve at most 1 and the AP is a function of the previously mentioned, the mAP has also the most desirable value of 1.

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i$$

IDF1 represents the ratio of the detections that were properly identified over the average number of ground-truth and computed detections. Similarly to the F1 score, the IDF1 score combines the identification precision and recall through their harmonic mean and in the original formula they show the explicit version. Ristani et al. (2016)

$$IDF1 = \frac{2 * IDTP}{2 * IDTP + IDFP + IDFN}$$

MOTA which stands for Multiple Object Tracking Accuracy is a widely used metric in object tracking that penalizes detection errors given by false negatives (FN),

false positives (FP) and fragmentations (Φ) normalized by the total number of true detections (T). Ristani et al. (2016) Milan et al. (2016) In a similar concept to previously described metrics, FN represents the sum of all false positives across the frames, namely all the times when the tracker detected a cell, but in fact there was no cell. Fragmentations refer to the switches of identity in a frame when the ground-truth does not present such change. The highest value MOTA can achieve is 1 and that is in the ideal case when there are no detection errors and the fragmentations are absent.

$$MOTA = 1 - \frac{FN + FP + \Phi}{T}$$

MOTP Milan et al. (2016) which is the abbreviation of Multiple Object Tracking Precision is the average overlap between all correctly assigned detections (true positives) and their ground-truths. In the formula, c_t means the number of matches in the given frame t, while $d_{t,i}$ is the overlap of the detection with its ground-truth.

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t}$$

3.3. Methods

3.3.1. Preprocessing

Due to the low contrast and difference in illumination in the raw images as seen in Fig. 7, the interphasic cells are very hard to detect. Therefore, we have used as a preprocessing step the CLAHE algorithm Yadav et al. (2014) for image enhancement.

Figure 10: Pre-processing of a B field image: (a) raw image, (b) result after the first CLAHE step, and (c) final result after the second CLAHE step

Figure 10 illustrates the pre-processing steps applied, from the raw B field image (a) to the first CLAHE step (b), ending with the second CLAHE step (c).

The comparison in Figure 11 shows that the three histograms, each corresponding to its left picture, significantly differ. The second histogram, corresponding to the image resulting from of a traditional histogram equalization performed by ImageJ, is better than the initial one in terms of contrast. However, it contains many values close to zero, corresponding to darkened areas of the image, effect not desirable for our application. We have chosen CLAHE as a contrast enhancement method because the traditional histogram equalization provided by ImageJ for example would extend the number of black pixels and thus confuse the upcoming algorithms.
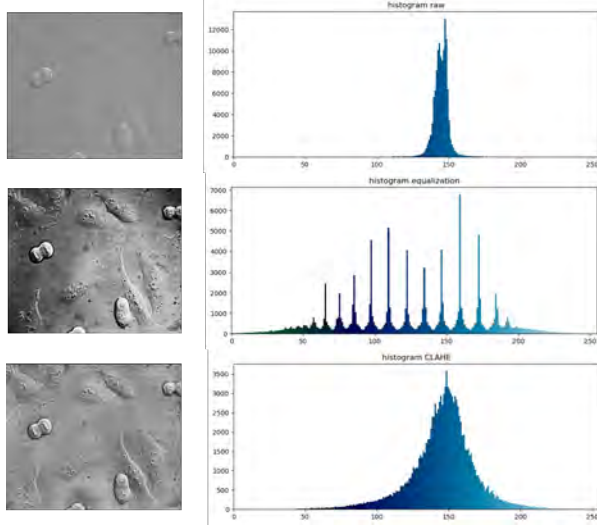
Figure 11: Image histograms for a B field image in its raw version (first row), histogram-equalised version (second raw), and preprocessed version using two CLAHE steps (third raw)

The third histogram, corresponding to the final pre-processed image, shows that we successfully enhanced the image contrast. We can also evaluate visually that the interphasic cells are more visible than in the original image.

3.3.2. YOLOv4

Introduced in Bochkovskiy et al. (2020), YOLOv4 is an object detection algorithm that achieved state-of-the-art performance on the MS COCO dataset. It utilizes as a base the DarkNet neural network framework which was written in C and CUDA.

In Fig. 12, we can see the main architecture of this algorithm of a general deep learning approach for object detection. Succeeding to the figure we explain the main features underlying architecture of YOLOv4.
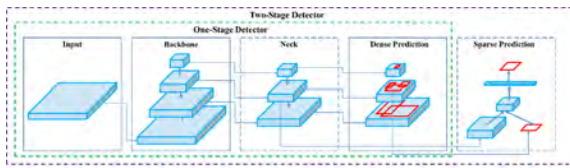
Figure 12: Object detector architecture

The input represents the raw images for the network. We have fed the network resized images as (384, 320) pixels which were also preprocessed using CLAHE.

For the backbone YOLOv4 applies Darknet53, which is a 53-layer convolutional neural network with the purpose of obtaining feature map.

The neck is used to enhance feature discriminability and robustness. For the YOLOv4 instead of FPN (Feature Pyramid Network) they use PANet (Path Aggregation Network) as a method of parameter aggrega-

tion from different backbone levels for different detector levels.

The head just as the neck is a subset of the backbone and it handles the predictions. Typically, the head can be divided into two types:

- one-stage detector: YOLO, SSD, RetinaNet

- two-stage detector: Faster R-CNN, Mask R-CNN

In the YOLOv4 architecture they have utilized YOLOv3 as the head.

In the backbone data augmentation was also employed. cutmix and mosaic, dropblock regularization and class label smoothing

They are also using a novel self-regularized non-monotonic activation function called MISH Misra (2019) that achieves great results in terms of performance and stability and which can be defined as:

$$f(x) = x * tanh(softplus(x))$$

When running the YOLOv4 code, we had to adjust some parameters to our custom dataset. The main modifications that we made in the configuration file are:

- Changed the batch from 1 to 64

- Updated the subdivisions from 1 to 16

- The width and height must be multiples of 32, therefore we set them to 384 and 320 respectively

- Set the number of classes to 2

- The value for maximum batches should be minimum 6000. In case of more than 3 classes, the value becomes the number of classes multiplied by 2000. Since we have two classes, our maximum batches is 6000 instead of 500500.

- The steps became a tuple of (80%, 90%) of the maximum number of batches. In our case, the steps became 4800, 5400

- Before each YOLO layer, we change the number of filters from 255 to the number of classes plus 5 all multiplied by 3. Hence, in our project the number of filters is 21.

Max batches leads to the total number of iterations for model training. Steps represents the number of iterations for which the learning rate will be multiplied by scale factor. Batch determines the number of images that are to be processed during one iteration while subdivisions mean the number of mini-batches in one batch, namely the number of batches that the GPU will process in one cycle.

3.3.3. YOLOX

YOLOX authored by Ge et al. (2021) is a 2021 object detector that achieved state of the art on the MS-COCO dataset providing improvements over its YOLO predecessors and provide a better trade-off between speed and accuracy.

YOLOX is using a YOLOv3 Darkent53 baseline and a and a Spatial Pyramid Pooling(SPP) layer.

Figure 13: Comparison between the YOLOv3 head and decoupled head

In Figure 13 the YOLOX authors show the YOLOv3 architecture of the head and the new architecture that they proposed with a decoupled head. This is a major unique difference between the latest algorithm and its ancestors.
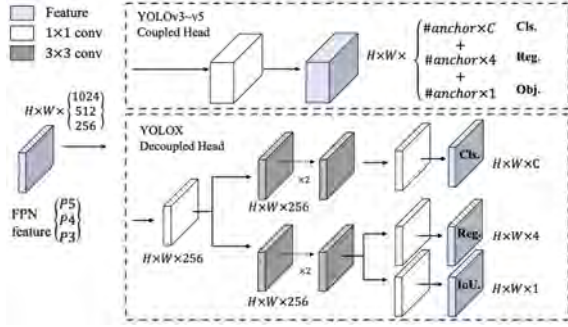
In order to increase the convergence speed they use a decoupled head that for each level of the Feature Pyramid Network the feature channel is subsequently reduced to a 1 x 1 Convolutional Layer followed by two parallel branches with 3 x 3 Convolutional layers for classification and regression tasks in the aim to reduce the conflict between the later two.

The improvements on the MS-COCO dataset Lin et al. (2014) compared to other models were the driving reason for us to utilize this method for our dataset. The results will be later discussed in the results section.

3.3.4. DeepSort

Tracking in the given dataset context is a challenging task even considering a perfect detection of the cell objects because unlike pedestrian tracking for example, our cells divide. Therefore, from supposedly one initial object will result two separate neighboring objects with different ids.

Considering that the state-of-the-art described in section 2 could not produce satisfying results, we have decided to look for an option outside the cell tracking paradigm and adapt it for our requirements.

DeepSORT Wojke et al. (2017) is highly performant multiple object tracking algorithm which had good results on the MOT challenge. An important asset of this algorithm is that it presented good results even for occluded objects. That is another reason why we have chosen to use this in the context of overlapping bounding

boxes of our cells when the wells become more populated.

SORT stands for Simple Online Real-time Tracking and although real-time tracking is not a necessary condition for our project, it could deem itself useful for future experiments for the biologists.

DeepSORT is based on five essential steps[1], also summarized in Figure 14:

1. detection of the objects;
2. update of the existing track positions via the Kalman filter;
3. grouping of the tracks by age and running the Hungarian algorithm on each cluster by the newest to oldest track;
4. processing the left unmatched and unconfirmed tracks of youngest age by the SORT;
5. setting the unmatched detections as new tracks.
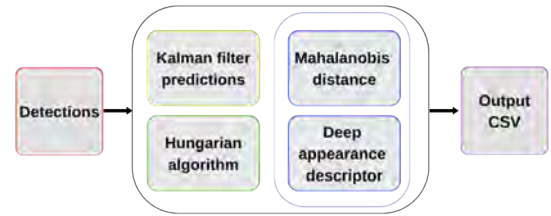
Figure 14: DeepSORT architecture

In our case, as we have previously explanied, we used the YOLO family of detectors. The detections alone are not enough to produce tracking, hence when a detection is multiplied from one frame to the next one, they use a linear constant velocity model. In the case that a detection is associated to a target, the detected bounding box is used to update the target state where the Kalman filter optimally solves the velocity components. In the situation where no detection is associated, the state is predicted only with the linear velocity model excluding the correction of the Kalman filter. In essence, it could be said that the Kalman filter is also providing the missing tracks.

It is also important to perform target association. To be able to assign detections to existing targets, each object's bounding box geometry is estimated by predicting its new location in its latest frame. The assignment is solved optimally by the Hungarian Algorithm, which has shown the particularity of showing very good performance when an object occludes another one. Furthermore, in our case when cells enter or leave the well, after a certain number of frames unique identities need to be created or destroyed from the current track. The algorithm computes the assignment cost metric as the intersection over union (IoU) between each detection and all predicted bounding boxes of the existing objects.

---

[1] https://papers.readthedocs.io/en/latest/tracking/deepsort/

In the paper Wojke et al. (2017) they display several formulas, out of which the following two describe the usage of the Kalman filter and the Mahalanobis distance.

Their square the Mahalanobis distance between the predicted Kalman states and the newly arrived measurements to incorporate the motion information to transform the goal into a problem solvable by the Hungarian algorithm.

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1}(d_j - y_i) \qquad (1)$$

In the above formula the (y_i, S_i) is the projection of the i-th track distribution into measurement space and d_j is the j-th bounding box detection.

The next formula considers the appearance space and evaluates the smallest cosine distance between the i-th track and the j-th detection. That is also important for us, because the cells have slightly different moving patterns throughout their lifespan in the well.

In addition, this step is useful is our context as well because the Kalman filtering estimation of the location is not precise enough since the images of the cells are taken 5 minutes apart, therefore in some cases certain cells will have a rather rapid displacement and the occlusions might happen at a fast rate.

$$d^{(2)}(i, j) = min\{1 - r_j^T r_k^{(i)} | r_k^{(i)} \in \mathbb{R}\} \qquad (2)$$

In the previous formula, r_j denotes the appearance descriptor computed for each d_j and k represents the track.

They combine the Mahalanobis distance useful for short-term predictions of object locations with motion and the cosine distance great for recovering identities after occlusion with less motion into one weighted sum as it follows:

$$c + i, j = \lambda d^{(1)}(i, j) + (1 - \lambda)d^{(2)}(i, j) \qquad (3)$$

To create trackers, any detection with an overlap less than the minimun IoU is considered to signal an untracked object. That is, if a bounding box B was detected, but it did not overlap enough with a present bounding box A, then the bounding box B encapsulates a new object. Initially, the velocity is set to 0. The tracker is subjected to a test period under which the target needs to be associated ith detections and thus the tracking of false positives is reduced. Tracks are stopped if they are not detected for a certain number of t_{lost} frames. In the case the object appears again in frame after that threshold, that object will be assigned a new identity.

To obtain the appearance feature vector a classifier is trained until a fairly good accuracy is obtained. Afterwards, the final classification layer from the network is excluded and the remaining dense layer produces a single feature vector to be classified. Then, nearest neighbour queries are used in the visual appearance to establish the previously mentioned measurement associations.

In the end, we visualize the video output of the tracked sequences and we export the results in a csv file containing the following format:

$$< im >< t_{id} >< cls >< x_m >< y_m >< x_M >< y_M >$$

This format represents in order the frame, track identity of the object, class of the object and the bounding box coordinates in the xmin, ymin, xmax, ymax format.

3.3.5. Lineage

We take the obtained CSV from DeepSORT and implement a post-processing step to be able to build the lineage. The procedure we propose is illustrated in Algorithm 1. If we find a cell that is mitotic and in the next frame the same cell becomes interphasic, it could mean that either a cell division occurred and the second is a daughter cell or initially a misclassification happened. We then check if the next cell with a new cell id is in the vicinity of the previously mentioned interphasic cell. In this case, the two interphasic cells are daughters of the mitotic cell in the precedent frame.

Furthermore, we output the total number of cells that were present in the observation lifecycle based on the total number of unique cell identities that were tracked.

4. Results

4.1. Detection

We have trained the models on 4 videos, namely fields A, C, D and E and tested on 2 videos. One of the test videos is similar to the training ones, while the other one is rather different from the training set. Below we show the comparison of the images.

In table 2 we show the results of the YOLOv4 results in comparison to YOLOX for both video B and T. We can deduce from the table that YOLOv4 achieves an average mAP of 76.02% slighly outperforming with 0.5% the 70.98% mAP of YOLOX.
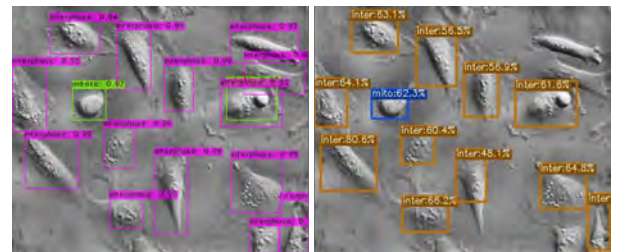
Figure 15: Comparison of predictions on a test image obtained using YOLOv4 (left) and YOLOX (right).

We see in Fig. 15 a comparison between the predictions resulting from the YOLOv4 algorithm in the left image and the YOLOX algorithm in the right.

```
for f ← 1 to max_frame do
    if cell.class[f − 1] is mitotic then
        if cell.class[f] is interphasic then
            if cellnewId.class[f] is interphasic & cellNewId.bbox[f] is inVicinityOf(cell.bbox[f]) then
                cell.parent[f] ← cell.id[f − 1] cellNewId.parent[f] ← cell.id[f − 1]
            end
        end
    end
end
```

Algorithm 1: Origin determination algorithm
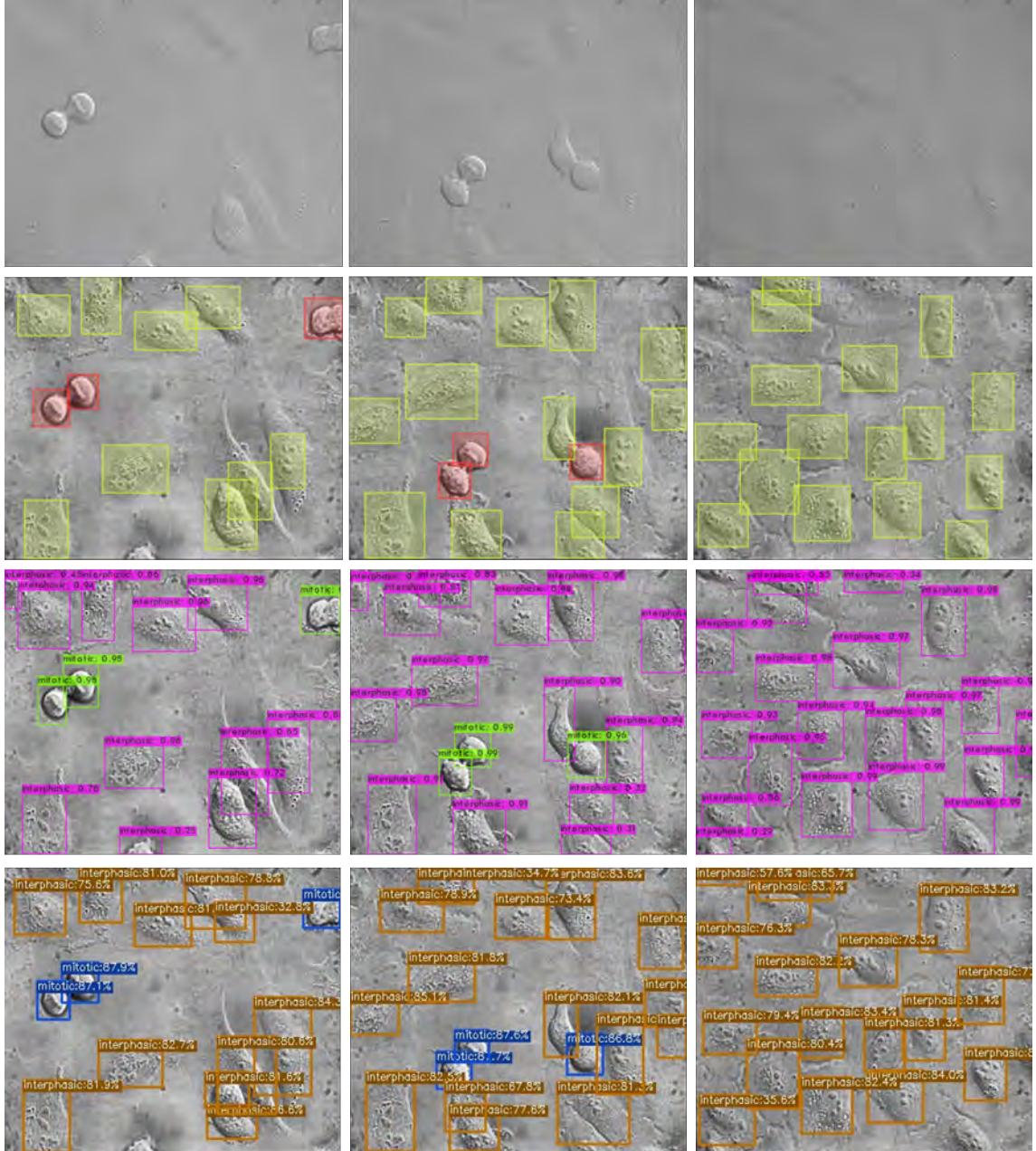
Figure 16: Comparison of test images from field B in raw format, annotations, predictions using YOLOv4 and YOLOX.

In Figure 16 we illustrate a comparison of test video B between the raw frames B02, B14 and B68, the corre-sponding annotations overlayed with the preprocessed image, the predictions of YOLOv4 on the third row and

3.10

Model	Batch	Video	F1	mAP
YOLOv4	16	B	0.79	0.8233
		T	0.78	0.6972
YOLOX	16	B	0.8	0.8779
		T	0.7	0.5418

Table 2: Comparison between different models on the test dataset

lastly the predictions of YOLOX on the bottom row.

4.2. Tracking

Video		R	IDF1	MOTA	MOTP
B	0.79	0.86	0.80	0.82	0.1
T	0.76	0.74	0.74	0.7	0.27

Table 3: Tracking results on test set

Table 3 shows the tracking results for half of each video from the test set according to the metrics a priori mentioned where P represents the precision and R the recall.

To sample visually the results of the tracking algorithm we show in the upcoming figure 5 consecutive frames from each video.

5. Discussion

Our journey in this project has in its core the honest desire to help biologists in their research to discover and develop better and new cures for highly impacting human diseases.

We have first sought information about the methods that were already implemented and we focused on the literature that was specific for transmitted light microscopy. Among them we have noticed that this field is indeed challenging and we can objectively state that DIC images are some of the most difficult ones and in great need of further research.

As at the beginning of the project we have worked on raw data and thought about the use of our results after the completion of the research period, we used the Ilastik tool that uses Machine Learning and was designed to achieve state of the art performance for cells.

Our approaches to use the Ilastik tool were explained in Chapter 2 as well as our final approach for this program. However, after after we completed our experiments we came to the conclusion that unfortunately this tool was not able to generalise for our dataset. Our initial aim was to provide a pipeline with minimal difficulty to use, but in this case Ilastik was not a solution.

We have then trained a U-Net with the manual segmentation masks for our dataset obtained by using Ilastik. We approached this method with the goal of obtaining other good segmentation masks that would be

used as input for the rest of the Ilastik pipeline. Yet, in spite of its previously known performance, for our dataset and with our pipeline it could not become a pillar in our solution.

Another possible trial was based on binarization by adaptive thresholding, gradient vector field followed by Gaussian filtering for nuclei detection to provide seed points for watershed which would serve for segmentation henceforth used for tracking as described in Li et al. (2010). In our case due to the faint contrast between the nuclei of the cell and background as well as the characteristic that the instensities of the content of the cells coincides with the background, this application was not suitable for our dataset.

In general, the mentioned algorithms and others that we encountered in the literature review performed fairly well on Phase Contrast cell images because of the halo effect around the cells which lead to better chances to drastically enhance the difference between background and cells, therefore leading to better segmentation.

Due to the size of the available dataset, the previously presented reasons and the tedious and time-consuming annotations to obtain segmentation masks, we have chosen to shift our approach to object detection followed by tracking.

We have chosen to test the performance of YOLO on our dataset because of its previous state of the art performance and fast inference time. We chose two models very different in implementation yet both highly performant: YOLOv4 and YOLOX and trained on four of our videos (A, C, D, E) and tested on the two remaining videos (B, T). Sample B is similar to the training data and the performance is naturally higher regardless of the model. On the other hand, the frames that construct video T consist rather different cells in regards to shape so the performance is lower in both models.

YOLOv4 is the older model out of the two and uses the Darknet framework. In order to run the experiment, we made use of the annotations that we created in YOLO format which consists of the following fields in this order: frame, class, x, y, width, height, where the later 4 are bounding box coordinates.

YOLOX is a 2021 method that brings several changes that we describe in section 3.3.3. To be able to perform out trials with this algorithm, we converted the YOLO annotations into COCO JSON format.

The results are rather close to one another. Yet, analysing the detections in particular for both classes, we decided to keep the YOLOv4 model for the upcoming step as it had the individual class AP closer in value compared to YOLOX. This was an important decision because the tracker needs good detections to keep the cell identities as consistent as possible.

For obtaining the detection results, we have set a confidence threshold of 25%. The reason behind this choice is that we have noticed that many cells are properly classified even though the threshold is low. Moreover, we
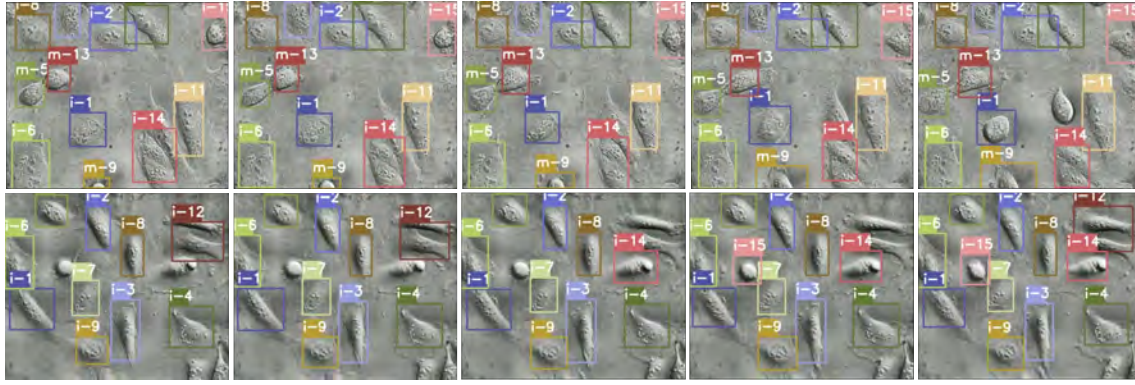
Figure 17: Comparison of tracking results for test fields B and T for five frames in a sequence

have observed that in general in the worst case a cell is either detected with the wrong class or is not detected at all.

In Figure 16 we show an example of the detections and we also comment on the metric in this case. Firstly, we notice a great difference in the contrast of the image and the improvement in terms of ease to detect cells even for the human eye. Secondly, we notice that in these frames as well as the overall experiment YOLOv4 is able to detect slightly better the cells. For example, in image B68 YOLOv4 detected the right lower corner cell whereas YOLOX did not.

When we were assessing the requirements of the project we have established that in general biologists do not annotate the marginal cells but it would have a neutral effect is those were to be detected. Consequently at the stage of creating annotations, we did not include marginal cells that were mostly not present in the frame our threshold being 50% presence. However as a result of the present marginal cells that matched the threshold and the similarity between cells, the algorithm learned to detect marginal cells as well.

The metric shows relatively good results, but in reality the output is better than the metric. The reason behind this statement is that the detected marginal cells are considered false positives, but in truth they present a good detection.

Moreover, to strengthen our previous comment about the confidence threshold we can observe for example in image B14, the cell located at the lower right border has a confidence of only 31%, yet it is well classified. Taking as example frame B02, we notice that the middle low cell has a confidence threshold of 25% and is classified as interphasic. In truth it is a mitotic cell, but from the image information alone and based on the training set it could arguably be classified as interphasic.

In order to track the cells we chose DeepSORT as an algorithm as we previously described in this paper. Not only does this method have a strong mathematical foundation, but it also has shown good results in combination with YOLO detectors.

In our results we can observe that the tracking is highly dependent on detection in the sense that a better detection leads to a better tracking result as well. Therefore, it was expected that the tracking accuracy for video B is higher than for video T.

DeepSORT has several particularities that we believe are worth mentioning. When the algorithm gets initialised, the first two frames of every video are lost. The reason for this lies in the fact that it utilizes those frames to use the YOLO weights to find the detections and then is able to locate them and then assign tracking identities to the cells. One major drawback is that some new identities have a too large magnitude between their id and the elder cells ids. Namely, somtimes a new cell can obtain for instance id 20 even though in the frame the highest id would be 15.

The tracking results are rather satisfactory, but it is important to take into account that the accuracy is prone to drop over time when the well becomes populated with more cells after several mitotic events took place.

6. Conclusions

In the present paper we have described our path to achieve our initial goal: provide a solution for cell detection and tracking.

The deep learning solution that we have initially proposed proved to be satisfactory and this approach can be generalised for similar cells. In a future case for other cell lines if the differences between cells are striking, new training can be issued as the annotation step with bounding boxes is highly sped up in comparison to traditional segmentation masks.

Another notable benefit consists in the comparison of our results with the DeepCell or Viterbi algorithms which could not track at all the images of the U2OS cell line.

6.1. Future Work

Research never stops, but rather it goes to the next step. That is why, in what regards future work we can

now output a few plans and suggestions based on our experience with this project.

Firstly, an immediate improvement could be achieved by correcting the assignment of new identities in the DeepSORT pipeline and increasing the efficiency of the lineage algorithm.

Secondly, more data should be annotated to obtain segmentation masks. Then a Mask R-CNN model could be trained and objectively evaluated based on the segmentation obtained of the cells present in bounding boxes.

Thirdly, other trackers could be assimilated into the pipeline and have the performance analysed such as: StrongSORT Du et al. (2022) which is the newest version of DeepSORT, CenterTrack Yang et al. (2021) or other centroid based tracker, Re3 which incorporates temporal information into the model Gordon et al. (2017) or an LSTM based tracker which would take into account memory of previous frames.

Fourthly, a tracking neural network plugin for Ilastik could be created. Another option to extend the usability of the project would be to build a standalone desktop application where new models could be trained and used to track the cells and build the cells' lineage.

Acknowledgments

References

Anjum, S., Gurari, D., 2020. Ctmc: Cell tracking with mitosis detection dataset challenge, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 4228–4237. doi:10.1109/CVPRW50498.2020.00499.

Asteriti, I., Di Cesare, E., Mattia, F., Hilsenstein, V., Neumann, B., Cundari, E., Lavia, P., Guarguaglini, G., 2014. The aurora-a inhibitor mln8237 affects multiple mitotic processes and induces dose-dependent mitotic abnormalities and aneuploidy. Oncotarget 5, 6229–6242. doi:10.18632/oncotarget.2190.

Baharlou, H., Canete, N.P., Cunningham, A.L., Harman, A.N., Patrick, E., 2019. Mass cytometry imaging for the study of human diseases—applications and data analysis strategies. Frontiers in Immunology 10.

Berg, S., Kutra, D., Kroeger, T., Straehle, C.N., Kausler, B.X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M., Eren, K., Cervantes, J.I., Xu, B., Beuttenmueller, F., Wolny, A., Zhang, C., Koethe, U., Hamprecht, F.A., Kreshuk, A., 2019. ilastik: interactive machine learning for (bio)image analysis. Nature Methods URL: https://doi.org/10.1038/s41592-019-0582-9, doi:10.1038/s41592-019-0582-9.

Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. URL: https://arxiv.org/abs/2004.10934, doi:10.48550/ARXIV.2004.10934.

Caldon, C.E., Burgess, A., 2019. Label free, quantitative single-cell fate tracking of time-lapse movies. MethodsX 6, 2468–2475. doi:https://doi.org/10.1016/j.mex.2019.10.014.

Damond, N., Engler, S., Zanotelli, V., Schapiro, D., Wasserfall, C., Kusmartseva, I., Nick, H., Thorel, F., Herrera, P., Atkinson, M., Bodenmiller, B., 2019. A map of human type 1 diabetes progression by imaging mass cytometry. Cell Metabolism doi:10.1016/j.cmet.2018.11.014.

Du, Y., Song, Y., Yang, B., Zhao, Y., 2022. Strongsort: Make deepsort great again. URL: https://arxiv.org/abs/2202.13514, doi:10.48550/ARXIV.2202.13514.

Gascoigne, K.E., Taylor, S.S., 2008. Cancer cells display profound intra- and interline variation following prolonged exposure to antimitotic drugs. Cancer cell 14 2, 111–22.

Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J., 2021. Yolox: Exceeding yolo series in 2021. URL: https://arxiv.org/abs/2107.08430, doi:10.48550/ARXIV.2107.08430.

Ghiasi, G., Lin, T.Y., Le, Q.V., 2018. Dropblock: A regularization method for convolutional networks. URL: https://arxiv.org/abs/1810.12890, doi:10.48550/ARXIV.1810.12890.

Gordon, D., Farhadi, A., Fox, D., 2017. Re3 : Real-time recurrent regression networks for visual tracking of generic objects URL: https://arxiv.org/abs/1705.06368, doi:10.48550/ARXIV.1705.06368.

Li, F., Zhou, X., Ma, J., Wong, S., 2010. Multiple nuclei tracking using integer programming for quantitative cancer cell cycle analysis. Medical Imaging, IEEE Transactions on 29, 96 – 105. doi:10.1109/TMI.2009.2027813.

Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P., 2014. Microsoft coco: Common objects in context. URL: https://arxiv.org/abs/1405.0312, doi:10.48550/ARXIV.1405.0312.

Magnusson, K.E.G., Jaldén, J., Gilbert, P.M., Blau, H.M., 2015. Global linking of cell tracks using the viterbi algorithm. IEEE Transactions on Medical Imaging 34, 911–929.

Milan, A., Leal-Taixe, L., Reid, I., Roth, S., Schindler, K., 2016. Mot16: A benchmark for multi-object tracking. URL: https://arxiv.org/abs/1603.00831, doi:10.48550/ARXIV.1603.00831.

Misra, D., 2019. Mish: A self regularized non-monotonic activation function. URL: https://arxiv.org/abs/1908.08681, doi:10.48550/ARXIV.1908.08681.

Moen, E., Borba, E., Miller, G., Schwartz, M., Bannon, D., Koe, N., Camplisson, I., Kyme, D., Pavelchek, C., Price, T., Kudo, T., Pao, E., Graf, W., Van Valen, D., 2019. Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning. bioRxiv URL: https://www.biorxiv.org/content/early/2019/10/14/803205, doi:10.1101/803205.

Ristani, E., Solera, F., Zou, R.S., Cucchiara, R., Tomasi, C., 2016. Performance measures and a data set for multi-target, multi-camera tracking. URL: https://arxiv.org/abs/1609.01775, doi:10.48550/ARXIV.1609.01775.

Skalski, P., 2019. Make Sense. https://github.com/SkalskiP/make-sense/.

Stringer, C., Wang, T., Michaelos, M., Pachitariu, M., 2021. Cellpose:

a generalist algorithm for cellular segmentation. Nature methods 18, 100–106.

Sullivan, K., 2001. Mitosis, in: Brenner, S., Miller, J.H. (Eds.), Encyclopedia of Genetics. Academic Press, New York, pp. 1224–1227. doi:https://doi.org/10.1006/rwgn.2001.0839.

Ulman, V., Maška, M., Magnusson, K.E.G., Ronneberger, O., Haubold, C., Harder, N., Matula, P., Matula, P., Svoboda, D., Radojević, M., Smal, I., Rohr, K., Jaldén, J., Blau, H.M., Dzyubachyk, O., Lelieveldt, B.P.F., Xiao, P., Li, Y., Cho, S.Y., Dufour, A.C., Olivo-Marin, J.C., Reyes-Aldasoro, C.C., Solís-Lemus, J.A., Bensch, R., Brox, T., Stegmaier, J., Mikut, R., Wolf, S., Hamprecht, F.A., Esteves, T., Quelhas, P., Demirel, Ö., Malmström, L., Jug, F., Tomançak, P., Meijering, E.H.W., Muñoz-Barrutia, A., Kozubek, M., de Solórzano, C.O., 2017. An objective comparison of cell tracking algorithms. Nature methods 14, 1141 – 1152. doi:https://doi.org/10.1038/nmeth.4473.

Vicar, T., Balvan, J., Jaros, J., Jug, F., Kolar, R., Masarik, M., Gumulec, J., 2019. Cell segmentation methods for label-free contrast microscopy: Review and comprehensive comparison. BMC Bioinformatics 20. doi:10.1186/s12859-019-2880-8.

Wojke, N., Bewley, A., Paulus, D., 2017. Simple online and realtime tracking with a deep association metric. URL: https://arxiv.org/abs/1703.07402, doi:10.48550/ARXIV.1703.07402.

Yadav, G., Maheshwari, S., Agarwal, A., 2014. Contrast limited adaptive histogram equalization based enhancement for real time video system, in: 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2392–2397. doi:10.1109/ICACCI.2014.6968381.

Yang, N., Wang, Y., Chau, L.P., 2021. Multi-object tracking with tracked object bounding box association. URL: https://arxiv.org/abs/2105.07901, doi:10.48550/ARXIV.2105.07901.

Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y., 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. URL: https://arxiv.org/abs/1905.04899, doi:10.48550/ARXIV.1905.04899.

Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D., 2018. mixup: Beyond empirical risk minimization. arXiv:1710.09412.

7. Appendix

While trying to optimize our results with Ilastik we have performed several trials to find the approach that would bring the most value to our research and raw dataset.
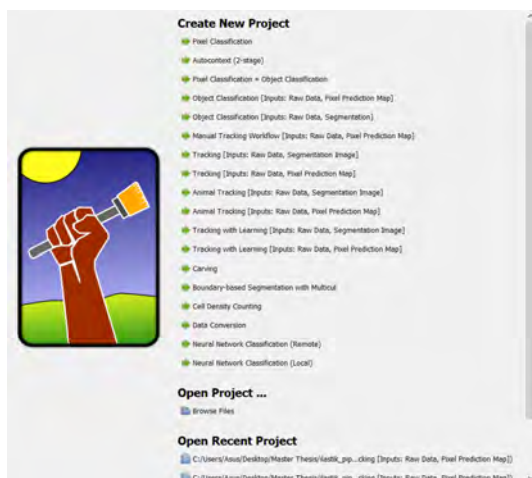
Figure 18: Ilastik Startup Screen

We have thereof began to use the Ilastik pipeline with

the Pixel Classification to generate segmentation masks and pixel prediction maps.

At first we have tried the standard trial where the background was drawn simply as as a few lines and the cells followed a very rough shape. Because of this, the algorithm was learning the wrong features and it could not properly produce further usable masks. Successively we tried the following approach:
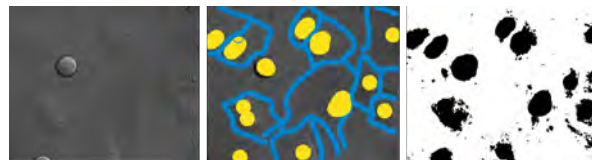
Figure 19: Ilastik second approach on Pixel Classification

Under this approach, we tried to encapsulate each cell in a rather rough background shape and draw the cell paying more attention to the countour of the nucleus. In this case we have use two classes: background drawn in blue and cells drawn in red.
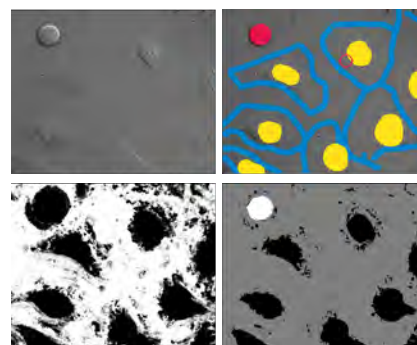
Figure 20: Ilastik third approach on Pixel Classification

The third hypothesis was to use 3 class segmentation from the beginning of the pipeline. Namely, 3 classes for pixel classification: background drawn in blue, interphasic cells colored in yellow and mitotic cells painted in red.

In the previous figure we can see the raw image in the upper left corner, the manual drawing of the desired delimitation of classes in the upper right corner, the resulting segmentation in the lower right corner and the resulting probability map in the lower left corner.

Although at first glance the segmentation map is better, due to the different versions in the ilastik workflow, specifically differences in the processing options between using segmentation or probability map as input, we have chosen to continue with the probability map as input.

For this approach we have reduced the distance between the background delimitation and the cell by drawing the cells closely encapsulated in the background. In the second image we can observe the yellow cells and the blue background pixel assignment. The result is still

3.14
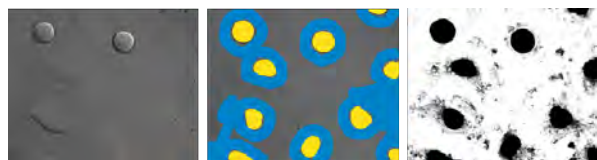
Figure 21: Ilastik fourth approach on Pixel Classification

showing misclassified pixels, but in this case the prediction is much better. A comment worth mentioning for further research is that we made use of the live prediction provided by Ilastik and we could correct the proper assignment of pixels on the go by first visualising the uncertainty map.

The outlier are not a major inconvenience at this step because they are sparse in comparison to the previous approaches and the probability map can be correct in the next step: object classification.
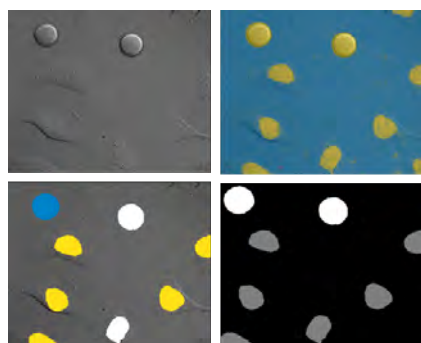
Figure 22: Object Classification in ilastik

In Fig. 22, we can see the raw image in the upper left corner, the threshold and size filter output in the upper right corner, the manual classification step in the lower left corner and the final object prediction in the lower right corner.

The threshold and size filter is a crucial step in the workflow as it helps clear the pixel prediction map previously obtained by the pixel classification. We can notice that now the cells are clearly differentiated from the background. Yet, one major downside is that the choice of the values of the threshold is global and there will be a trade-off between a clear image and keeping all the cells present.

After setting the threshold for segmenting the objects, we classified them into mitotic and interphasic. In this case, the mitotic cells are colored in blue and the interphasic ones are colored in yellow. After training the model for several frames, we have left two objects unclassified manually as we can see in the lower left image. However, we can see in the lower right image that the classification has been done correctly and the upper two cells are classified as mitotic and the rest as interphasic.

In order to use the tracking workflow, we use the

masks obtained from the object classification step. The previous step served as a thresholding and object map creation step.

Figure 23: Tracking with ilastik preprocessing

In Fig. 23, we can see in the left-most image the thresholding and size step. Because we have very carefully drawn the cells and masks before in the steps prior to tracking, this step was not entirely needed, but it serves as a double check of the cleaning of the map. We have chosen the hysteresis method for thereshold and specified not to merge objects. This way certain very close separate cells could be differentiated.

The middle image shows the division detection step where we label each cell as either not dividing in yellow or dividing in blue. In the workflow this is labeled as an optional step, but we have deemed this step as mandatory for our data.

The rightmost image show the object count classification step where we can input and afterwards correct the prediction of a single object depicted in blue or two objects painted in red.

Figure 24: Thresholding step in tracking

We must mention before moving forward the impact of the thresholding step. In this case the threshold was slightly different and the separation step had a different outcome. Therefore, the cell that was supposes to be identified as a whole, became halved. Such undesirable outcome shows in the tracking map as well.
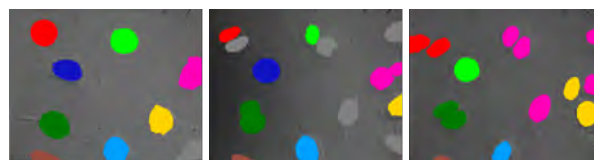
Figure 25: Tracking with ilastik frames 9, 10, 24

In Fig. 25, we present visually the tracking output of ilastik. Each color represents a tracking id assigned to that specific cell.

A positive aspect is that the lower green cell changed shape and entered mitosis and was able to maintain the

same id throughout the frames. In addition, the brown
and the light blue cells remained constant in annotation.
The red mitotic cell underwent division and even though
in the subsequent frame the daughter cells received an-
other id, in later frames this was corrected.

On the negative side, the upper dark blue cell lost its
identity throughout the frame. The bright green mitotic
cell from frame 9 divided and by the 24th frame the
daughter cells had been assigned a completely different
id. The lower yellow cell inherited the id of the pink cell
even though no division happened to the yellow cell nor
the displacement was very large. A new cell has entered
the frame and was assigned a new id in frame 10, but by
frame 24 it was detected as a daughter of the pink cell.
Moreover, after the pink cell present in frames 9 and 10,
throughout the frames

Recently, a new version has been released that con-
tains a plugin which allows the usage of a pretrained
network in the BioImage Model zoo[2] format in order to
perform pixel classification.

Unfortunately a common trait between the ap-
proaches was that although with enough training it per-
forms relatively well on the trained images, for our
dataset this tool is not able to properly generalise.

Subsequently, we tried to find another solution and
we tried to obtain segmentation masks using a U-Net
pretrained on masks obtained from Ilastik.

In addition to our mentions in Section 2, we would
like to add here that although the U-Net has shown rela-
tively good results for 3 classes (interphasic, mitotic and
background), it could not generalise well for other im-
ages. which in the long run it would prove to be coun-
terproductive as the acquisition of new annotated data
would be very time consuming. In addition to this, it
had worse results for two classes (interphasic and mi-
totic).

Figure 6 contains the U-Net prediction for the image
present in the left in comparison with the groundtruth
present in the middle in the binary class trial. After-
wards, figure 26 illustrates the result of the prediction
for a 3 class case where the interphasic cells are black,
the mitotic one is white and the background is gray.
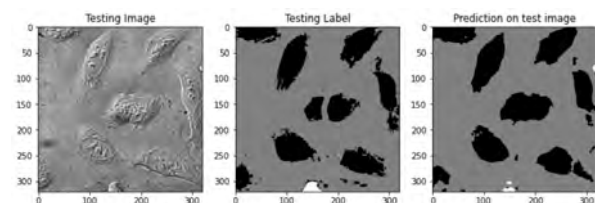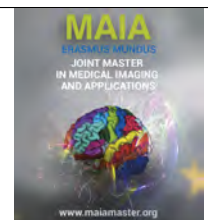
Figure 26: U-Net prediction with ilastik masks

---

[2]https://bioimage.io/#/about

Automated Abdominal Aortic Aneurysm Detection on CT Scans

Anwai Archit, Bram van Ginneken

Diagnostic Image Analysis Group, Radboud University Medical Center, The Netherlands

Abstract

Computed tomography (CT) scans enable the detection of local enlargements in the abdominal aorta (AA), resulting to straight-forward quantitative and qualitative understandings, typically instated as abdominal aortic aneurysm (AAA). Although, the segmentation of aorta is disposed to stall in presence of expanded lumen or intraluminal thrombus as a result of insufficient spiteful examples, raising the susceptibility for uneventful outcomes of an aortic rupture. The motion of this research proposes to develop and validate a fully automated deep learning algorithm to segment and measure AAAs on abdominal CT scans. The computer-aided detection (CAD) model is steered by a self-configuring convolutional neural network (CNN), which plumps for essential decisions in a standardised environment to design the 3D segmentation pipeline, regardless of the dataset diversity in the domain. It uses an additional 3D instance-based vertebral segmentation software bundle for independent vertebrae labelling. It coheres with a post-processing routine to perceive the growth patterns by investigation across the aortic centerline around strong anatomical landmarks. It benefits from supplementary measurement techniques of the maximal diameter and cross-section area for gaining extensive insights of the main characteristics of AAA. The system evaluates the relationship between the AA and vertebra level surface features. Conclusively, it generates a portable document, devised to group the anticipated aneurysmal information.

The 3D CAD system agrees with expert's suggestions about the existence of the aneurysm in 398 institutional images, exhibiting a high capacity to generalize across genders and portions of a full body CT scan using solely radiologist-supported quantitative speculations from the radiology reports. The end-to-end routine achieves an 95.7 % dice score coefficient (DSC) on the validation subset for patient-specific cases, indicating a modest agreement with radiologists within an average difference of 0.3 cm in the relative measurement of maximal AAA diameter, thus justifying the possibility of generalizing to the detection of aneurysms using report-based textual information only.

Keywords: computed tomography, abdominal aorta, abdominal aortic aneurysm, vertebra, deep learning, convolutional neural network, computer-aided detection, segmentation, detection, centerline, post processing, portable document, dice score coefficient

1. Introduction

In the western world, abdominal aortic aneurysms (AAA) are commonly associated with increasing incidences of morbidity and mortality among the elderly population. In clinical research, the popular discussions on the therapy (namely, endovascular aneurysm repair, EVAR) of ruptured AAA (rAAA) takes into account non-modifiable risk factors such as male gender, advanced age and inherent genetic features. The incidence of AAA is much higher in men (7.6%), however women (1.3%) are susceptible to aggressive aortic enlargement pattern and behold higher risks of rupture (Li et al., 2022; Pleumeekers et al., 1995). The perils of dilation in the largest vessel in human body crossing the threshold of the normal arterial wall diameter are fatal in cross-gender studies, leading to an asymptomatic rupture mortality of 85–90% (Kent, 2014). With each decade, the patient-specific risk of AAA increases significantly for men over 50 years old and women between 60 and 70 years old (Bengtsson et al., 1992). Although the therapeutic stratification suggests the prevalence of AAA among men is four times higher than among women, and among people with family history of the disorder

Figure 1: The overview of aorta. *(left to right)* 3D rendering of the surface anatomy corresponding with the aorta, and the appearance of aorta as the overlays in three respective planar views, axial plane, coronal plane and sagittal plane on CT scan

is four times higher than among those without a family history, smoking still forfeits the rest as the strongest modifiable risk factor (Kent et al., 2010).

The flexible identification of multi-sized aneurysms is the pinnacle of effective clinical diagnosis and surveillance for the AAA. While abdominal ultrasound (US) and computed tomography (CT) angiography (CTA) are the most commonly used diagnostic imaging tools for AAA detection and helping to foresight pre-operative and post-operative decision-making and planning, magnetic resonance imaging (MRI), positron emission tomography-computed tomography (PET-CT) and incidental detections are also in practice (Wanhainen et al., 2019). Despite the highly sensitive (95%) and specific (100%) nature of ultrasonography (Fleming et al., 2005), CT stands out to be the exquisite choice of the abdominal region for aortic aneurysm detection outlining the precise anatomical information, as shown in Fig. 1 (Hansen, 2016; Landman et al., 2015). Even so, plethora of studies and standard guidelines still emphasise on the ardent need of invariant descriptions of surface context to help the contextual investigation of the ailments. The importance of abdominal aorta (AA) surface markers for determining the likely beginning of AA further till the bifurcation of the AA becomes significant at this point. (Ali Mirjalili et al., 2012). The surveillance of the growth of aneurysm becomes crucial when exceeding 50% of the average aortic diameter, giving rise to perceptive monitoring of small AAAs (3.0cm–5.4cm) and prophylactic actions for patients prone to rupture ($\geq 5.5cm$)(*Chaikof et al.*, 2009).

The advent of principled techniques for AAA identification has expanded the scope of research in medical image analysis, providing a sincere insight to benefit the physicians with comprehensive qualitative and quantitative analysis.

2. State of the art

Convolutional neural network (CNN)-based approaches have provided encouraging research in the realm of computer-aided diagnosis or detection (CAD) throughout the last decade (Gao et al., 2019). Modern machine learning techniques provide extensive insights of the expanse and morphology of aneurysms, which

is vital for automated characterization of AAA in CTA (Raffort et al., 2020).

Throughout the history of screening high-resolution medical images, pinnacle of traditional image segmentation methods underlined by Raffort et al. (2020) confers the striking achievements of active shape model segmentation scheme by de Bruijne et al. (2004), triangular mesh-based graph search model by Lee et al. (2010), level set method by Zohios et al. (2012) and many more. Caradu et al. (2021) supports the point by presenting a comparable qualitative assessment of a fully automated software (PRAEVAorta) for infraneral aneurysm detection comparing with physician's reproductions on 100 scans, but here and now leading proposals using deep learning. Some researchers summit across open discussions of the influence of regularization methods like Otsu's thresholding and K-means clustering in penalising output probability maps by CNNs to magnify the segmentation quality (López-Linares et al., 2018b).

Many studies suggest minor rearrangements in CNNs to extend its algorithmic utility. Lu et al. (2019) anchors on 3D-UNet (Çiçek et al., 2016) for 321 scans and appraises the largest axis of detected ellipses (Fitzgibbon et al., 1996) to counter the overlooked incidental detection on 57 examinations, achieving 91% sensitivity and 95% specificity. Golla et al. (2021) verifies the deep convolutional networks (3D ResNet) using layer-wise relevance propagation on 106 scans to achieve an area under the curve (AUC) of 0.971. Dziubich et al. (2020) manifests the calibre of ensemble of end-to-end convolutional neural networks (U-Net, ResNet, VBNet) to outmatch standard methods unaccompanying any post-processing step, by 3% on Dice metrics. Brutti et al. (2022) propose a 2.5D-based approach to merge spatial information in 2D fusion step and minimize the computational requirements compared to 3D networks to achieve eminent results.

By bringing *DetectNet* and DCNN to clinical foregrounds with comparable findings, López-Linares et al. (2018a) highlights the importance of non-contrasted thrombi segmentation. To reach a stirring F1-score of 91.97 % for thrombus detection, sequential detection and segmentation tracks may be topped up with optimized loss functions to counter problems like class im-

balance (smooth L1 loss and modified focal loss, respectively) enunciated by Hwang et al. (2022).

Several evolutionary studies have endorsed mixed-effect models in conventional follow-up data utilizing multi-modal information (in-silico or omic dataset) for effective detection of AAAs. Jiang et al. (2020) proposes a two-step training of deep belief networks (DBN) that encapsulates geometrical features with growth and remodelling (GR) models based on finite element method (FEM) and adjusting the pretrained weights to capture the aneurysmal features in the second step. Hong and Sheikh (2016) spotlights the intrinsic ability of DBNs to learn in presence of a smaller training sample and lower segmentation complexity.

For the screening and monitoring of patients in AAA risk management, Sokol and Nguyen (2022) stresses on the use of a multi-parametric scheme that includes therapeutic risk factors from electronic health record (EHR), anatomical references from ultrasound (US) and CT, and genetic details of the variants to jaw the inherent risks.

Habijan et al. (2020) confirm their take on deep supervision in 3D U-Net across the decoder tract together with deconvolutional layers replacing the upsampling layers, giving rise to the average DSC of 91.03%.

Mohammadi et al. (2019) canvasses a proposal of CNNs to detect the appearing ailment and categorized quantitative estimation of its severity using hough circles algorithm to estimate aortic measurements. Finding the maximal diameter using a fully automated pipeline by establishing the medial axes of lumen, centerline extraction and detection of maximum equivalent diameter rivets a rather pragmatic approach to assort the aortic ellipses (Adam et al., 2021; Brutti et al., 2022; Lareyre et al., 2021).

In matters of acquiring statistical understandings from available descriptions of the bulge in AA, there is thin limelight on one of the most important abdominal surface landmarks clearly visible in CT scans, the vertebral levels of the AA. Ali Mirjalili et al. (2012) appreciates the clinical significance of the levels of vertebra for therapeutic distinction of anatomical regions of interest, despite huge disparities in the combinations of ethnicity, age, BMI, gender and possible medical conditions for over 108 CT cases. The research study also shows the statistical association of celiac trunk at T12 vertebral level (>40%), extending AA until the bifurcation commonly at L4 (60%), backing the common medical incidences with standard discussions. To explore the importance of precise anatomical segmentation and identification of vertebra to gather understandings about the peak correlation amidst AA along the different vertebral levels, Lessmann et al. (2019) proposes an iterative instance-based segmentation model with an optimised traversal strategy across the vertebral column to obtain favourable labels.

Although research results of modern day CNNs tends

to break the grip of hurdles in biomedical image segmentation, most of them fail to generalize on fresh datasets beyond the presented tests. Finally, Isensee et al. (2020a) conceive the possibility by proposing a completely automated, adaptable U-Net based architecture called nnU-Net that configures pre-processing requirements and network parameters confidently with cost-effective design choices. The publicly accessible tool has demonstrated impressive results on prominent public datasets in the biomedical area by automatically managing broad ranges of hyperparameter adjustments based on intrinsic data information.

2.1. Contributions

The goal of this study is to bring together the most up-to-date state-of-the-art methodologies for creating an automated narrative of end-to-end 3D learning for AAA identification and quantitative studies on CT images. The following overview summarizes contributions of our experiments:

- We investigate a customized self-configuring CNN architecture (nnU-Net) that focuses on retrieving key structural discriminators in CT images on top of textural identifiers across distinct 3D patches of the full volume. Regardless of intravenous (IV) contrast, manifolds of thoracoabdominal cavity were employed to make the most out of data diversity-based segmentation. We manually edit the annotations for each patient assigned to the training subset to include the missing aortic sections, if necessary.

- We acquire the aorta segmentations on inference set and lead the key-points for centerline extraction, which can be used to determine effects of local aortic mesh enlargements. We look at the local descriptors and see how they help with the post-processing scheme.

- We utilise a pre-trained instance-wise segmentation network to extract the vertebral levels for the CT volumes. The research percepts to make use of case-specific labels by combining them with forthcoming quantitative information to help clinicians draw inferences from a statistical curvature-based pattern in AA along the vertebral column.

- To introspect the aneurysm, we compute diameters across the centerline with reference to acquired points in 3D coordinate plane, compute the respective maximum diameters and cross-sectional areas, and compare the evaluation to the patient's radiological diagnosis report with the real world. We create a non-linear trace of diameters compared with the relative change of distance vector traversing upwards, starting from one aorta end point to the other for better discussion in our study. In

contrast to the centerline-based diameter extraction from the aortic mesh model, we use binary masks in the axial plane of CT scans to locate the circular blobs of interest, offering two-dimensional macrodescriptors along the slices, and check the output values to put our method to test.

- We compare our descriptive performance to that of homogenous research limited to monotonous test cases using 398 distinct cross-gender evidences from the university medical facility. We direct our model to create a portable document, the reference report for the patients, and we hence test its ability to detect AAA on CT scans in general.

3. Material and methods

3.1. Dataset

The first share of the dataset includes 50 clinically collected abdominal CT scans (Fig. 2) from a colorectal cancer chemotherapy study and a retrospective ventral hernia investigation at the Vanderbilt University Medical Center (VUMC). The key objective of the open-source dataset is to furnish pseudo-labels for the internal dataset. It is supported by precise annotations drawn by the field experts for thirteen organs and evaluation of the accuracy of volumetric labels facilitated by the radiologists (Landman et al., 2015). The raw data is divided into 30 (60%) training samples and the rest 20 (40%) for testing. Besides the availed test subjects, the training set is randomly partitioned into 6 (20%) validation subjects across the forthcoming five folds of learning.

The cornerstone of this veritable study involves the internal dataset, comprising of 398 CT scans (Fig. 3) from the Radboud University Medical Center (RUMC). It is succeeded by pseudo-labels, obtained during the inference stage from the VUMC dataset. The authors eye down the pseudo-labels to observe some future line of actions, and take necessary modification measures for 9 (2.3%) distinct patient-cases (discussed in Section 3.1.3). Special attention has been paid to inspect and consider harmonic subjects of interest for the learning arrangements of the networks.

3.1.1. CT Scans

A series of 50 CT scans have been taken from VUMC during the portal venous contrast phase with variable volumetric dimensions (512 x 512 x 85 – 512 x 512 x 198) and field of views (approximately 280 x 280 x 280 mm³ – 500 x 500 x 650 mm³). The slice thickness goes from 2.5 mm to 5.0 mm, and the in-plane resolution extends from 0.54 x 0.54 mm² to 0.98 x 0.98 mm² (Landman et al., 2015).

Meanwhile, 398 CT cases have been internally assorted by the Department of Radiology and Nuclear Medicine at RUMC, by the application of natural language processing (NLP) on the radiology reports by competent authorities (in Dutch) from the cohort of 2000 to 2021. The gathered cross-study cases have a huge variability in the volumes, dimensions belonging to either of the two among 512 x 512 or 1024 x 1024 along the dense axial plane (Volumetric Depth {mean depth : 561, interquartile range (IQR) : 34-2014}) with voxel depth ranging from 0.25 mm to 5.0 mm, and a voxel spacing stretching from 0.30 x 0.30 mm² to 0.97 x 0.97 mm². The patients are aged men and women (Female {median age : 68, IQR : 44-89}, Male {median age : 70, IQR : 43-117}) accompanied by a speculated medical association with abbreviations like "**AAA**" (one of the key factors for casting the obvious) from their respective radiology reports (prevalence : 1.5%). The experts discover many false positives (FP) as an outcome of minor typographical mistakes in the radiological scripts. For instance, "***AAAnhoudende*** exacerbaties van bronchiectasieen met subfebriele temp geen verwekker kweek." translates to "Sustained exacerbations of bronchiectasis with subfebrile temperature no causative agent culture." is one of the many FPs likely lost in translation. Although the earmarked example is a simple spelling mistake of *AAAnhoudende* instead of *Aanhoudende* with a literal translation to *Continuous*, the reproduces have a good chance to hold clerical errors in the different descriptive sections of the medical reports, leading to small inconsistencies in the choice of the target cases. Accordingly, all the chosen volumes and their textual counterparts have been validated carefully by the authors to consider sufficient dilated cases of aorta.

Prior to training, all the scans undergo spatial resampling to the median target spacing of the dataset. Further discussions in Section 3.2.1 will unveil the side-by-side automated techniques. All things considered, the training volumes are resampled using the third-order spline-based interpolation and the respective ground truths are treated with linear interpolation (Isensee et al., 2020a). Regardless of the deceit, research investigations have been playing with non-rigid registration across the volumetric cross-sections (Landman et al., 2015). Yet, the primary objective of the studies envision to capture the anatomical features.

3.1.2. Clinical Annotations

The VUMC CT scans have been assessed by clinical specialists using the MIPAV utility (McAuliffe et al., 2001) for validating (and adjusting, if required) the multi-organ annotations put together by a team of experienced undergraduate volunteers (Landman et al., 2015). All the following multi-organ annotations are conditioned to convert into a foreground class of aorta and the rest as background. The authors review the binarized masks for the 30 VUMC cases to find out the potential chances of missing the delineations for the aor-
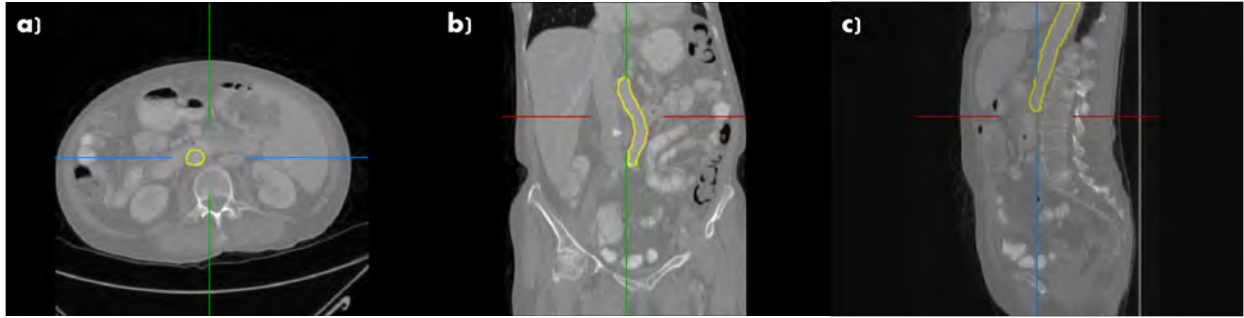
Figure 2: The CT scans for a single patient from the VUMC dataset is shown above. The instances contain aorta segmentation with yellow overlays in three respective planar views, a) axial plane, b) coronal plane and c) sagittal plane.
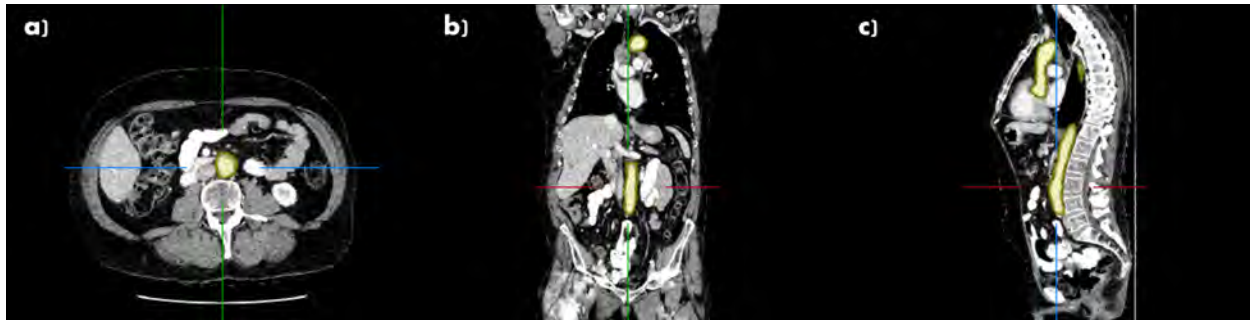
Figure 3: The CT scans for a single patient from the RUMC dataset is shown above. The instances contain aorta segmentation with yellow overlays in three respective planar views, a) axial plane, b) coronal plane and c) sagittal plane.

tic arc and/or the ascending thoracic aorta in majority of the volumes (frequency : 60%), to prepare for the subsequent hurdles and ways to consider the essentials.

In the RUMC CT dataset, the responsible radiologists have performed the scans with the consideration of medical guidelines for the need of intravenous agents in the study in order to pinpoint the origin of unexplained symptoms. Fig. 2 and Fig. 3 portray the diversification across the volumes and the existence of typical or dilated aorta in both the datasets.

3.1.3. Manual Segmentation

Keeping the absent aortic annotations in mind, the demand for handcrafting the paired pseudo-labels in the internal dataset is the preliminary step. The inherited dataset with voxel-level annotations for each VUMC CT scan is expected to train for generating aortic segmentations using a three-dimensional CNN (elaborated in Section 3.2.1). The bunch of optimized trained model weights are handled in the inference scheme of the detection network for the RUMC dataset to result in pseudo-labels for the aortic region.

Hereafter, the authors rectify the discussed hurdles and update the masks using the interactive MITK (Wolf et al., 2005) tool to include the missing delineations in different aortic regions (prominently, the intraluminal thrombus, aortic arc and ascending thoracic aorta) into the predicted labels, as revealed in Fig. 4. The overlay masks aligned over the CT volumes are refactored using

the *Add* and *Live Wire* functions in the *2D Tools* menu of the *Segmentation* module. On occasions, the *Subtract* and *Erase* functions have been explored to remove the avoidable annotations.

3.2. Model Architecture

To reckon the extent of an aneurysm formed in the AA alongside its respective anatomical surface landmarks, we employ two parallel 3D CNN bottlenecks (N_1, N_2) followed by a post-processing implementation (P_{AAA}). The sequential model (succeeded by the training of VUMC dataset using the imminent network for benefiting the other dataset) is guided by the detection network - N_1, for the purpose of segmenting the lumen and thrombus, if present in abdomen, backed by different measurement techniques to quantify the detected aneurysms. Conversely, another pre-trained instance-based segmentation model - N_2 predicts the vertebra levels from the incoming inference set in isolation. A part of post-processing routine takes over the calculation of the center of gravity for vertebral labels along the spine towards individual planar localization. Multi-planar CT volumes across various structural regions are processed using patches of 3D convolutions. The prevailing discussion about a series of study in the RUMC scans having dimensions of 1024 x 1024 returns with a solution destined to adapt the higher resolutions to the observable dimensions. The N_1 pipeline anticipates to adapt these into the automated pipeline by resampling

Figure 4: Abdominal cross-section along the axial plane on the CT scan. a) Original region of interest - presence of intraluminal thrombus b) Pseudo-labels shown by red overlay maps c) Updated delineations using the MITK tool shown by yellow overlay maps d) Superimposing b) and c) to illustrate the qualitative difference

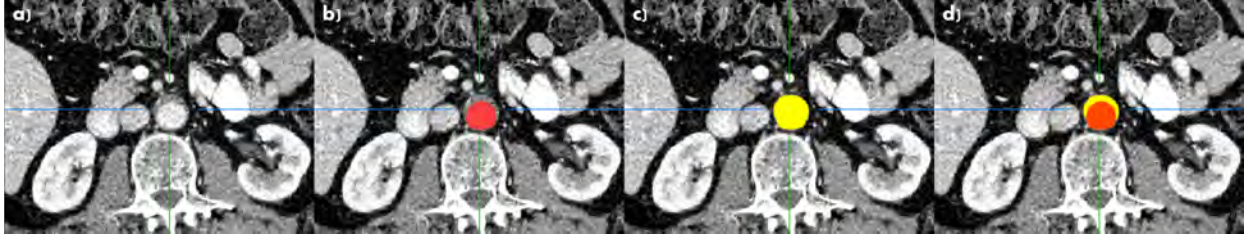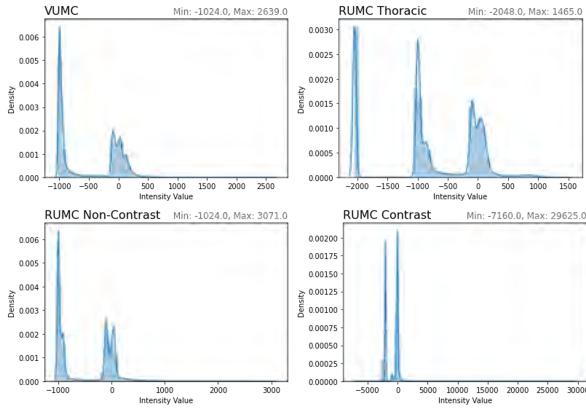Figure 5: Intensity distributions of CT scans from different anatomical regional with variable presence of contrast

the aforesaid cases to twice its pixel spacing in order to scale all the three dimensions with new spacing. Intensity normalization is performed thereafter on resampled input volumes (512, 512, x, where x is varying depth of the volume) using global dataset percentile clipping and z-score with global foreground mean and standard deviation to handle the huge intensity range, as traced in Fig. 5. The investigated U-Net-like model N_1 adapts its topology for utilising the encoder-decoder structure with skip connections, instance normalization, leaky ReLU and deep supervision (Isensee et al., 2020a). At the training schedule, N_1 uses sum of cross-entropy and dice loss as its loss function. Ultimately, binary predictions from N_1 and multi-label segmentations from N_2 are individually forwarded to the queue of P_{AAA}. As a result, end-to-end supervised learning concludes by drawing statistical inferences about the possible enlargements supported by report-guided measurements from the internal dataset. All output parameters are aggregated into a single extensive report for expert's reference. The entirety of the end-to-end mechanism is illustrated in Fig. 6.

3.2.1. Detection Network

The central module of our study is the detection network (N_1) or nnU-Net (Isensee et al., 2020a), carefully ornamented with all the evidences stitched together in

Fig. 7. The dominant purpose of N_1 is to generate highly sensitive voxel-level segmentations of aorta on CT scans.

An infrarenal AAA, the most common of its kind, covers an estimate ellipsoidal volume of 140 ± 70 cm^3 and is the usual representation of catastrophic enlargements (mean maximal diameter : 5 ± 1.0 cm) (Renapurkar et al., 2012). The automated selection of input patch size for the N_1 training schemes of VUMC dataset is decided at 192x192x48 voxels per volume and 160x128x112 voxels per volume for RUMC dataset. The overlapping strategy throughout the entire volume utilizes adjoining information of the local anatomy, grading larger patch sizes above the batch size (resolved at two for both N_1 training instances). Patches of designated shapes paired with their labels are extracted from all across the volume to train N_1. The core of this research is followed by the curse of severe class imbalance, hence oversampling is implemented by the robust pre-processing strategy of nnU-Net. 66.7% of patches are selected from random locations within each of the training samples, while 33.3% of patches are guaranteed to seize the foreground classes present in selected training sample. The training of N_1 uses the combination of soft dice and cross-entropy loss :

$$\mathcal{L}_{total} = \mathcal{L}_{dice} + \mathcal{L}_{CE} \tag{1}$$

For deep supervision outputs at all resolutions, a corresponding downsampled ground truth segmentation mask is used for loss computation, estimated as sum of the losses ($\mathcal{L}$) :

$$\mathcal{L} = w_1 \times \mathcal{L}_1 + w_2 \times \mathcal{L}_2 + ... \tag{2}$$

where, weights are reduced by half with a decrease following in each resolution and normalized to the sum of one : $w_{i+1} = \frac{1}{2} \times w_i$

The nnU-Net uses credentials of classical U-Net (Çiçek et al., 2016; Ronneberger et al., 2015) as its base three-dimensional architecture using a configurable topology as per the number of downsampling operations on each axis depending on patch size and voxel spacing. The creditable design choices of the N_1 pipeline using a set of heuristic rules guides the reflec-

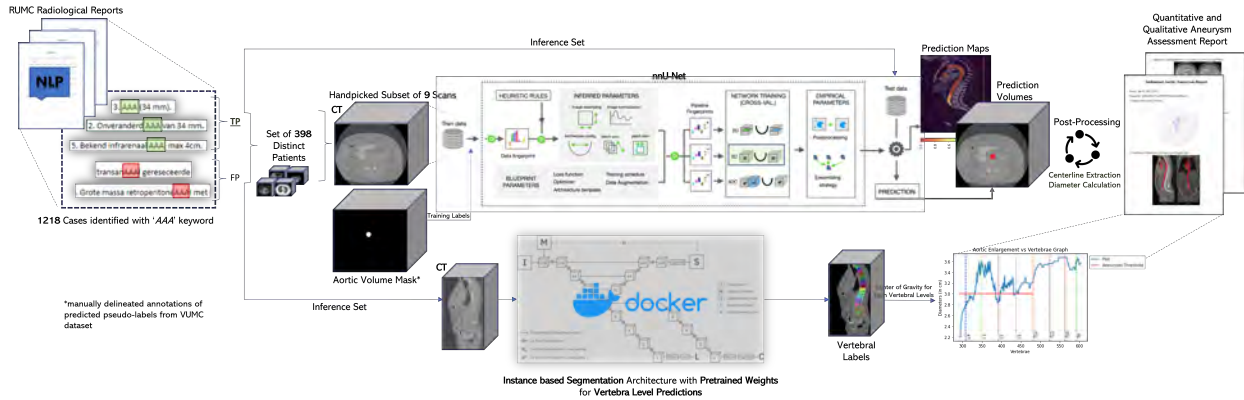Figure 6: End-to-end automated abdominal aortic aneurysm detection pipeline

tion of substantial improvements in the paradigm of semantic segmentation within biomedical imaging applications. The 3D U-Net empowers the strengths of globally capturing multi-resolution features from the human anatomy. N_1 uses two convolution blocks per resolution step composed with instance normalization for normalizing contrast between spatial elements on single samples (Ulyanov et al., 2017) and leaky ReLU nonlinearity to enable negative slopes in both encoder and decoder networks. The upsampling operation is performed using convolution transposed while the downsampling operation is implemented using strided convolution. Amidst the trade-off between performance and memory usage, considerations of initial number of feature mappings is decided at 32 and doubled (or halved, depending on the bottleneck) with each downsampling (or upsampling, relying on the change in feature maps) operation. The number of feature maps is also restricted to 320 for the 3D U-Net to tone down the final model size. Thereafter, N_1 is trained with deep supervision as well, allowing gradients to be laced down deep into the network and facilitates training of all layers in the network by adding additional auxiliary losses in decoder to all but the very two lowest resolutions. The built-in architecture of the 3D U-Net model is shown in Fig. 8.

A five-fold cross validation is observed for training the entire U-Net configuration with default choice of 1000 epochs per fold. The network weights are optimally learned with stochastic gradient descent, Nesterov momentum ($\mu = 0.99$) and initial learning rate of 0.01 decaying throughout training using the polynomial learning rate scheduler. For batch size as small as two, the number of foreground patches after handling class imbalance is forged to a minimum of one, making the quintessential decision to input one random and one foreground patch for the N_1 training per batch. For elevating the performance of the strong foundations, diverse data augmentation techniques are applied by means of rotations, scaling, influence of gaussian noise and gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirror-

ing (Isensee et al., 2020b).

3.2.2. Predictive Modelling of Instance-based Vertebral Segmentation

The goal of the segmentation network, N_2, is to provide specific vertebra level predictions using instance-to-instance segmentation of each scan for cross-referencing the AA. The outputs are utilised by P_{AAA}, taking the segmented vertebral column into account for locating centers of gravity of individual vertebra levels, focused on the lumbar and thoracic vertebrae. This helps clinicians to develop an understanding between the correlation of the local enlargements across the abdominal vertebral landmarks.

Lessmann et al. (2019) proposes a four-component vertebra-by-vertebra segmentation and labelling method (N_2) based on a fully convolutional neural network (FCNN) (Fig. 9) convened to execute many tasks at once by analyzing the patch size of 128 x 128 x 128 voxels, big enough to contain a minimum of one vertebra. The input volumes are resampled to an isotropic resolution of $1mm$ x $1mm$ x $1mm$ to avoid divergent predictions on cases with varying resolutions. The patches are sampled in such a way so as to ensure the compulsory presence of vertebral bones besides the 25% of random sampling across the entire volume. Along the patches, the iterative inference network segments a unique vertebra, and the respective anatomical information is shared about the succeeding vertebra near by for employing the patch to shift for segmentation of the next vertebra. The network takes inspiration from the traditional 3D U-Net architecture composing skip connections and default number of convolutions with padding and batch normalisation in the encoder-decoder arrangement (Çiçek et al., 2016; Ronneberger et al., 2015). Binary classification of all voxels in the patch happens to segment the voxels from a 3D patch, enhanced with an instance-based memory that informs N_2 about the already segmented vertebrae. This allows segmentation of voxels corresponding to a single instance rather than all vertebrae visible in the

Figure 7: Automated self-configuring heuristics for the nnU-Net pipeline

Figure 8: The 3D architecture of nnU-Net for the RUMC training dataset
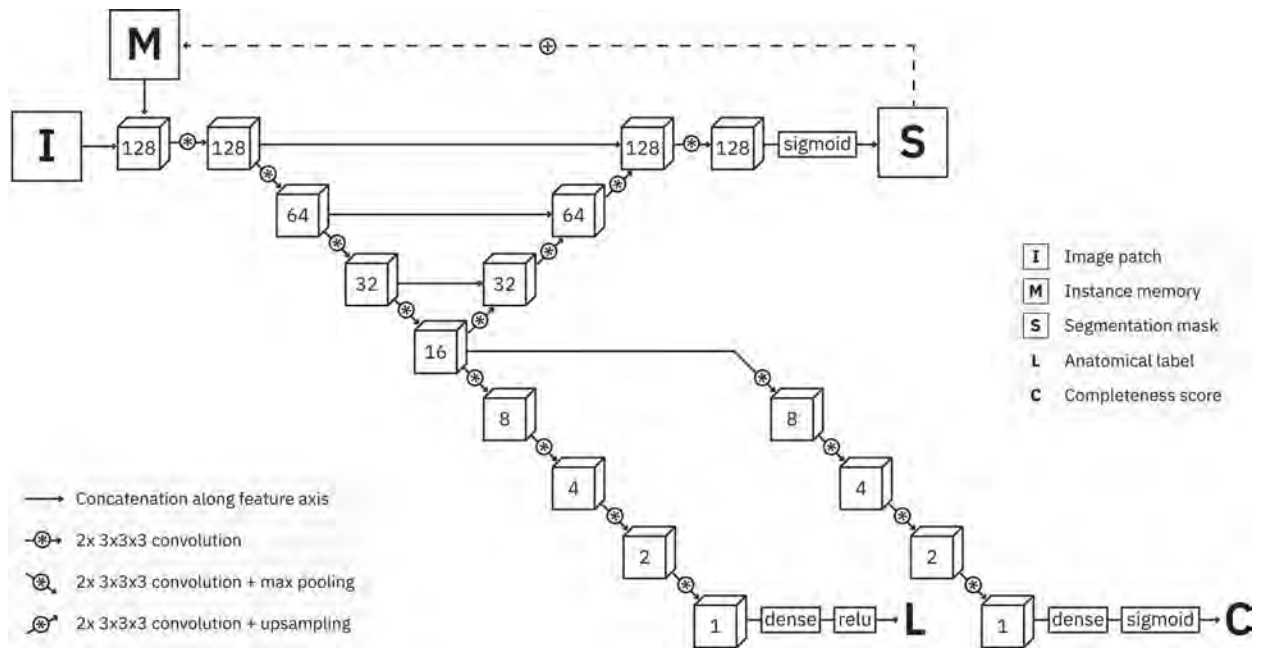
Figure 9: Instance-based vertebral segmentation and labelling
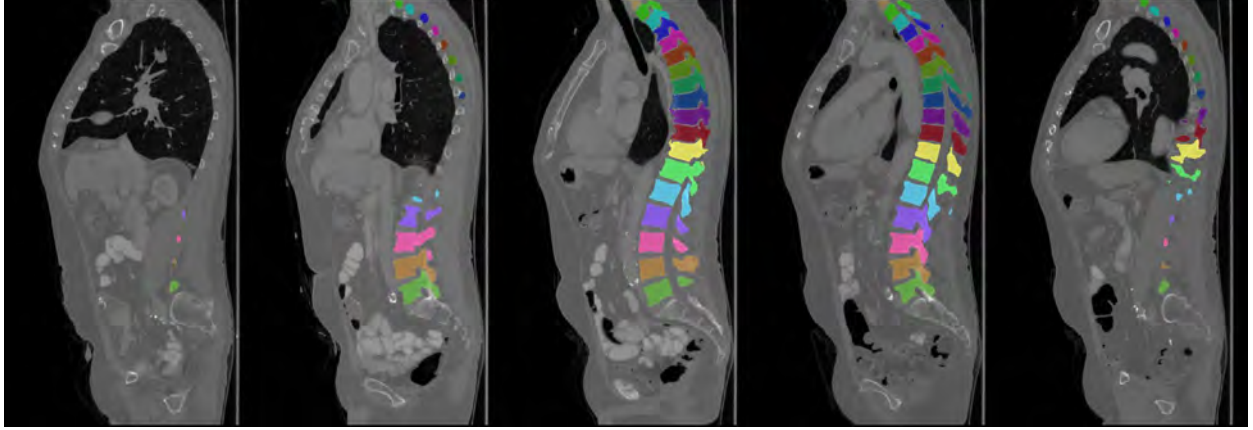
Figure 10: Vertebrae segmentation and multi-label classification outputs of the vertebral levels represented along the sagittal plane

patch. This binary flag per voxel information is utilized by N_2 to segment just the vertebrae that haven't been segmented yet. When a vertebra is fully segmented, the instance memory is updated, causing the network to disregard that vertebra in next iteration and instead focus on upcoming vertebra. The identification sub-network is the third component, predicting anatomical label of each of the identified vertebra levels using maximum likelihood estimation. A completeness classification sub-network is introduced to the network as the fourth component to distinguish between fully visible and partially visible vertebrae. In return, P_{AAA} takes just the fully visible vertebra levels into consideration. The N_2 network architecture is illustrated in Fig. 6. The loss term λ is the combination of segmentation error used to minimize weighted number of incorrectly labelled voxels, labeling error uses $l1$ norm difference of predicted (p_L) and true labels (t_L) and the completeness classification error used to define the binary cross entropy between the true labels (t_C) and the estimated probability of complete visibility (p_C).

$$\mathcal{L} = \underbrace{\lambda \times FP_{\text{soft}} + FN_{\text{soft}}}_{\text{Segmentation error}} + \underbrace{|p_L - t_L|}_{\text{Labelling error}}$$
$$+ \underbrace{(-t_C \log p_C - (1 - t_C) \log(1 - p_C))}_{\text{Completeness classification error}}$$

where:

$\lambda =$ weighted factor of the cost

FP_{soft} and $FN_{\text{soft}} =$ differentiable expressions for respective number of predictions

The research constructs upon interconnected communication between the N_2 model architecture and pretrained model weights along with well-defined preprocessing and post-processing channels for the prediction pipeline packaged into a software bundle, popularly known as *containers* (Merkel, 2014). This light-weight container is chained in the entire execution with forthcoming P_{AAA} tools to reward the quantitative measurements around significant vertebral landmarks. The predictions on the inference set provide us with valuable information about the vertebrae, as shown in Fig. 10.

3.3. Post-Processing

Characterization of aortic aneurysms based on quantitative and qualitative assessments on predicted binary masks is the principle objective of the post-processing pipeline, P_{AAA} discussing patient's expected treatment phase. The vision of trained nnU-Net networks is elucidated by softmax prediction maps outputted by N_1 in finding the vulnerable aortic surface walls. On the other hand, N_2 gives us more information about various vertebral labels, helping us to establish the anatomical range of AA for precise AAA identification beyond the aneurysmal threshold. The post-processing approach takes full charge once the analogous inference models (N_1, N_2) deliver the desired prediction outputs. The pipeline ensures to capture characteristics of potential abdominal aneurysms, allowing exploration to gather conclusions and contemplate with immediate therapeutic actions. For universal convenience of comprehending bulges in aorta, wide range of geometrical descriptor-based discussions support the evaluations of an aneurysm using aorta diameter, cross-sectional area of the expansion, volumetric changes in aorta, mechanical surface modeling of aorta, and so on. The study condenses a wide variety of statistical findings into easy parametric realisations of estimating the ruinous range of diameter, which will be useful to radiologists. The final connection is based on comparison of report-guided radiological outcomes, visualised endorsements of the aneurysm and quantitative findings from the automated algorithm.

3.3.1. Conventional Methods

Most of the quantification algorithms rely on simplified shape extraction using macrodescriptors (maximum incircle, geodetic length) and statistical lengths (Feret, Martin and Nassestein diameter) along different planes of visualization (Kroell, 2021). This study perceives to assess the baseline descriptors with final automated processing module.

On one hand (complemented with a parallel experiment on centerline extraction using statistical descriptors, briefly discussed in Appendix A) the pipeline builds a rational sequence of diametric calculations using the characteristics of elliptical pixel arrangements along the axial plane. Because of the commonly associated challenges in the curvature of thoracic aorta to find measurements solely on the basis of shape descriptors, the approach now excludes the region and evaluates thoracoabdominal and abdominal surfaces, by rejecting slices with multiple blobs. The next move aims to detect circular blobs on the inverted binary masks (foreground as 0 and background as 1, satisfying the default detector settings) relying upon carefully chosen parameters for optimal detection based on extracting the entirety of the connected components and returning the length of keypoints. We further stir previous discussions on algorithms by Kroell (2021) for statistical measurements based on macrodescriptors of detected blob(s) to return the central position and diameter of the maximum inclosing circle, the largest possible circle that touches the projection area from the inside. The algorithm iterates over the possibility of determining the diameter of circle based on spotting new candidate points surrounding the center of mass of detected blob, aiming to settle the constraints to output the optimal measure (Li et al., 2020).

The research tries to inculcate a rather pragmatic approach to extract lengths of major axis (M_{major}) and minor axis (M_{minor}) of the detected ellipse. We find the average of two chords to approximate the measurement of diameter, resulting to a negligible difference (disregarded further) compared to the diameter from the maximum inclosing circle, as shown below :

$$d = \frac{M_{\text{major}} + M_{\text{minor}}}{2} \qquad (3)$$

3.3.2. Centerline Extraction

After assistance in the prediction step in obtaining aortic segmentations for the various RUMC CT instances, the automatic post-processing pipeline starts its bit for centerline extraction once the prediction results are in. We start by visualising the 3D aortic masks and decide on necessary reconstruction of the isosurface using marching cubes algorithm (Fedorov et al., 2012; Lorensen and Cline, 1987). Thereafter, number of points in the mesh model is smoothed by topology-preserving reduction of surface triangles and merging

of coincident points, removal of unused points (i.e. not used by any cell), treatment of degenerate cells is made possible to consider necessary percentage of useful points (Taubin et al., 1996). The pipeline continues to automatically select centroids of the endpoints for the aortic mesh by first selecting a relative position of the starting point with respect to origin of the mask and it's respective endpoint. This mechanism can be manually performed as well using 3D-Slicer tool and interactively mark endpoints on mesh models at ends of the aorta for precise centerline extraction, reducing geometrical and computational complexity. While the requisites are dealt with, centerline model with the respective centerline curve points in the RAS coordinate system are extracted using *Extract Centerline* module in the vascular modeling toolkit (VMTK) (Antiga et al., 2008). The entire process has been summarised with an illustrated example in Fig. 11. The algorithm returns a JSON file indexing the respective centerline points (i.e., an array of lists having three coordinate values per index) which is now available to be utilised for the calculation of maximal diameter.

3.3.3. Diameter Extraction

The algorithm uses *Cross-Section Analysis* module of 3D-Slicer tool to provide necessary quantitative measurements for lumen and respective dilation in diseased cases (Fedorov et al., 2012). With input as the processed mesh surface model with centerline curve points in possession, it provides us with measurements of minimum inscribed spherical (MIS) diameter and surface area for the maximum and minimum aortic cross-sections for individual curve points in RAI coordinate system. We brush aside the automated measurements of the extremes as they tend to locate minima near endpoints, and maxima in thoracic aorta. From here, another function works upon the diameter column and provides the statistical assessment of the incremental pattern of diameter in the aorta. For this purpose, it uses the starting point of the centerline at the aortic annulus in the thoracic region until the beginning of the branching of iliac arteries. It also computes the length of the centerline by the cummulative sum of all the consecutive distance vectors, which we use further for correlation. An additional exploratory step of extracting 3D voronoi diagram is implemented by construction of convex polygons along the dense center-points of aorta, giving us an insight of the effective distance of surface with respect to centerline (Antiga et al., 2003). The jitters resemble larger polygons built across aortic regions with a relative enlargement and color scheme showing traces of the direction of maximum descent. The complete structure of centerline extraction has been illustration in Fig. 11. The major passed down takeaway is the non-linear graphical plot of the change in calculated aortic diameter along the centerline.

| a) Binary Mask | b) Mesh Model | c) 3D Centerline Curve along the end nodes | d) Centerline Extraction | e) Voronoi Diagram |
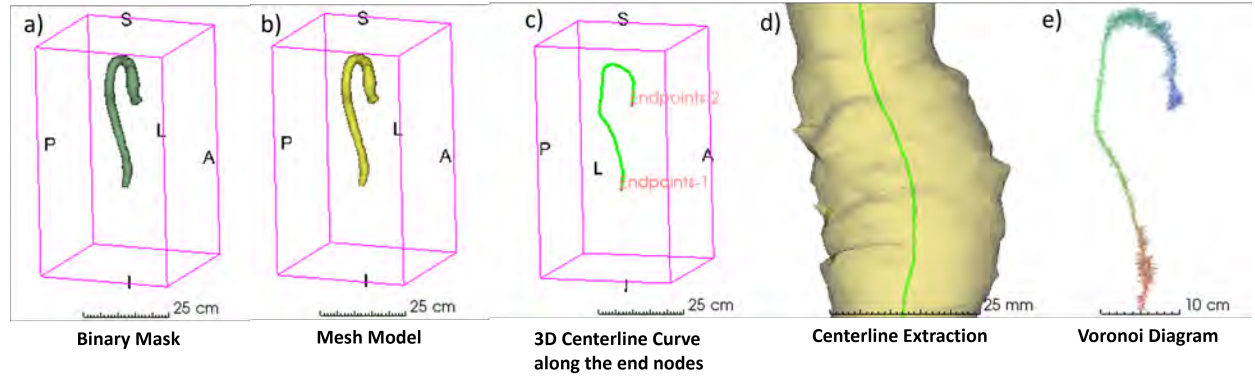
Figure 11: Visualisations of the VMTK output modules for the centerline extraction pipeline

3.3.4. Graphical Analysis of the Abdominal Aorta
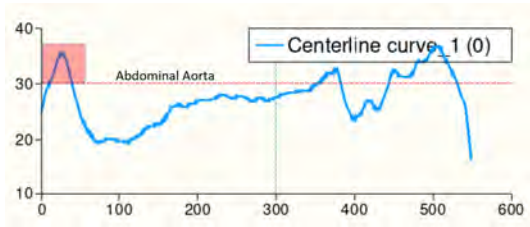
Figure 12: Graphical quantification of AAA with respect to the end-point distances (centerline points)

The final component must be handled now that all of the pieces are in place. The research confirms the legitimacy of basic techniques for calculating diameter, taking into account the overall desire for lowering complexity and enhancing process efficiency. The centerline extraction approach, which makes use of VMTK's automated modules, gives us a better understanding of the dense centerpoints traversing along the centerline to report the changes in diameter over the entire aorta (Fig. 12). Although most methods focusing just on AA have been developed as a part of process for grasping the region of interest along distance vectors of the centerline, further examination into spatial relationship with anatomical markers is needed to demonstrate the use of end-to-end automated intuitive diagnostic tools.
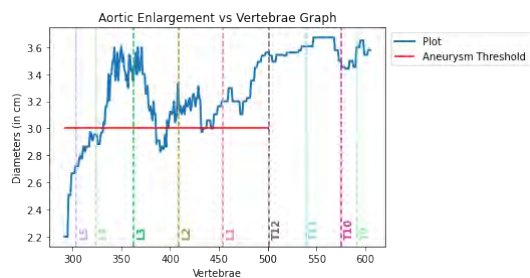
Figure 13: Graphical quantification of AAA with respect to the vertebral column, thresholded from T12 and summing up at the detected end of the lumbar spinal region

In this scenario, the study supports vertebra identification and uses a mathematical analogy to maximize the benefit. Following the heuristics developed under P_{AAA}, N_1 and N_2 pipelines provided separate predictions on the same patient cases. The geometrical analysis starts with binary masks that must be present in the predictions, as well as the reciprocal labelled values of segmented vertebrae and computed diameters. The practical necessity to examine the obtained diameter along the axial plane with reference to vertebrae becomes part of our discussion by combining the arrays of data. Secondly, initial ideas of constructing a scatter plot of diametric measures along individual labels, while simple, would not be the greatest explanation for a definitive revelation of the aneurysmal curve. This necessitates determining the center of gravity (shown in Algorithm 1) of each vertebrae and connecting the landmarks as a simple scalar representation on a non-linear plot of the observed diameters and axial slices.

Algorithm 1 Pesudocode to compute the center of gravity of each unique vertebra label

---

Require: *unique_vertebra_labels*
  for element i in *unique_vertebra_labels* **do**
    Computer *start* which is the slice where i begins
    Computer *end* which is the slice where i ends
    center_of_gravity = (*start* + *end*)/2
  end for

Clear explanations concerning the presence of AA around lumbar spinal area (L1–L5), with AA commencing at lower terminal ends of thoracic spinal levels (T12), have been addressed, largely to get to this point of asserting our introspection, as per set standards and guidelines (refer to Section 2). Finally, a graphical depiction of non-linear curve with quickly changing diameter throughout the axial plane in thoracoabdominal region is obtained, indicating aneurysmal threshold with a significant focus on the vertebra levels behind the AA (Fig. 13).

4.12

3.3.5. Portable Document

The final heuristic of the pipeline introduces a portable document format (PDF) document generation under python using the outputted variables and independent screenshots from the executed algorithm. The choice of library has been made to enable easier formatting options with automatic text justification, page and line breaks, and plethora of support for links, colors and images, best suited for our study.
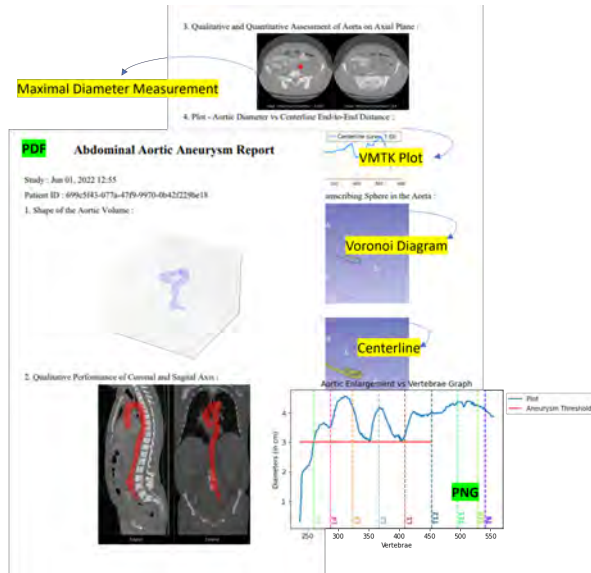
Figure 14: Diagnostic reference reports and graphical quantification of the AAA with respect to the vertebral column along the axial plane

In the aforementioned trail, it compiles outputs of extracted centerline, generates pair of screenshots along coronal and sagittal planes with complete overlay maps of aortic binary masks over median slice number along respective axes, generates another pair of screenshots along axial plane finding maximum and minimum diameter slices to have better understanding of underlying aneurysmal structure. It is completed with the nonlinear curve of aortic diameter along the centerline, illustrating subsequent changes in diameter within the aorta. Each report mentions the time of the study and the patient filename to help users affix the case-specifc outcomes with necessary theurapeutic discussions. For the purpose of anatomical visualization, the 3D aorta is represented with the voronoi diagram and aortic mesh overlayed on the centerline model (also shown in Fig. 11). Along with a PDF output (as shown in Fig. 14) of the document, the algorithm outputs a portable network graphics (PNG) for the vital plot (Fig. 13).

4. Results

4.1. Qualitative Assessment of Detection Networks

Simple CNN implementations supervised by precisely delineated aortic masks provide decent segmen-

tations for the undilated aorta to the common population. In presence of deviations in target predictions, the essential parameters for therapeutic screening are usually missed. Fig.15 portrays the aforesaid revelations in the prediction outcomes of our implementation on the RUMC inference dataset. It reciprocates the importance of case-specific delineations (as the measures taken into consideration in Section 3), showing decent qualitative improvements in localization of enlargements in aorta. We notice that N_1 training on VUMC dataset captures the strong contrast of intraluminal region, beholding useful information to guide inferences through aortic segmentation. With additional exploration of manual interventions in succeeded pseudo-labels by considering the thrombus and missing aortic regions in the thoracic cavity, distinction of dilated aorta tends to increase the scope of detection network. While we initially attribute our assessment to contrast-enhanced CT scans, it is important to mention the appreciable results on inference set regardless of the anatomical cross-section and intravenous trace in abdomen. Despite the facts mentioned, the pipeline misses to detect massive enlargements above 6 cm. It follows similar trails in recently ruptured aneurysms and some post-operative instances with familiar incidences of huge dilations in the AA.

4.2. Quantitative Assessment of Detection Networks

The intrinsic post-processing module of 3D nnU-Net automatically looks after metric calculations for validation samples. Over six data samples in cross-validation, N_1 network obtains an average validation Dice Score Coefficient (DSC) of 0.935 for entire aorta segmentation in VUMC cases, right from aortic annulus in the ascending thoracic aorta to the bifurcation of iliac arteries at the end of AA. The learning curve for this dataset in Fig.16 speaks for itself. There is a strong overlap between loss function curves suggesting the limited, yet worthwhile reusability of the optimized weights. While pre-processing criterion shortlists two cross-validation data samples per fold in RUMC cases, average validation DSC stands at 0.871 identifying and segmenting the intraluminal thrombii from the vast stretch of contrast. Fig.17 seems to have random perturbations across the learning curve, with possible plateauing in the learning outcomes. From the Table 1 and Table 2, we introspect quantitative performances of initial network across the five-fold cross validation by nnU-Net. Although both have comparable DSC as modern-day research investigations, it turns out that the rejection rate for false samples across both volumetric validations is quite high. We notice a significant variability in performance across the training scheme for both datasets. The jaccard's index for foreground class of RUMC validation set proves to be higher, while managing to include missing aortic regions compared to the other. The mean scores of precision and recall for respective dataset breathes the efficiency of trained weights for RUMC dataset, equivalent
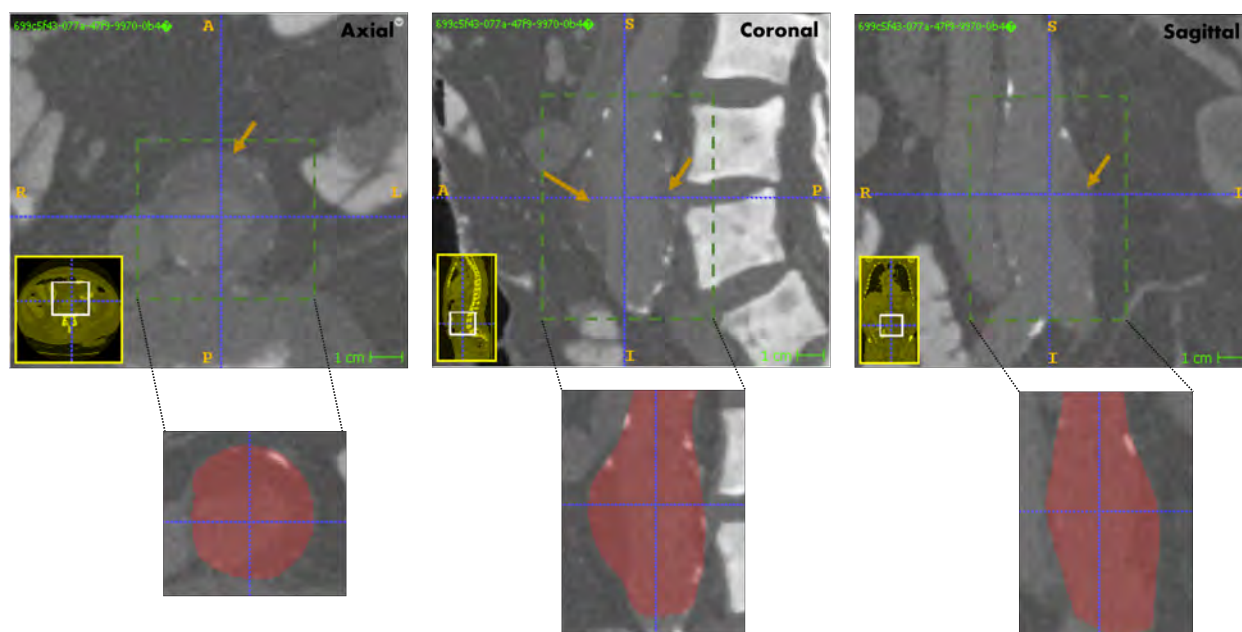
Figure 15: Qualitative assessment of the detection network on the RUMC inference set
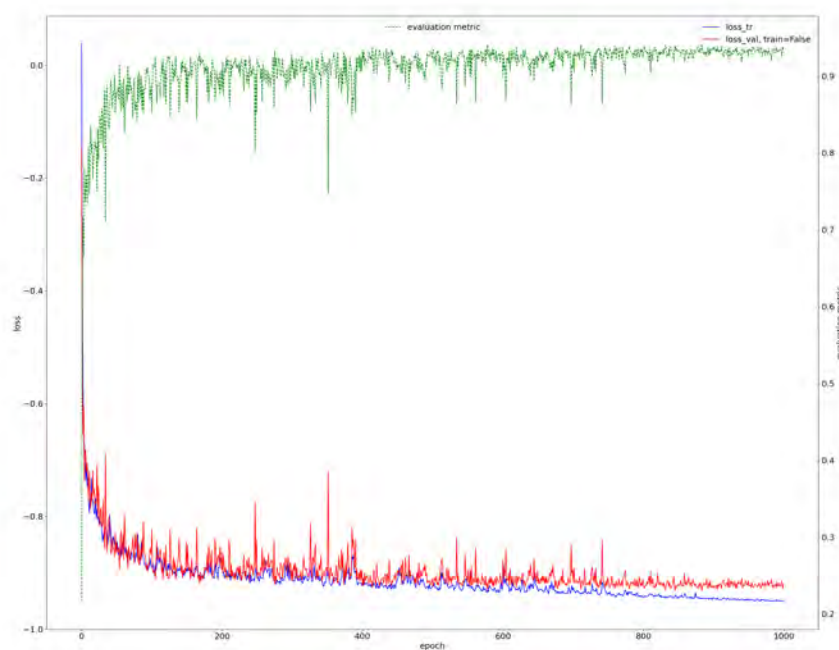
Figure 16: Per-epoch progress of Dice score coefficient, training loss and validation loss for the 2nd training fold - VUMC train-val plot

4.14
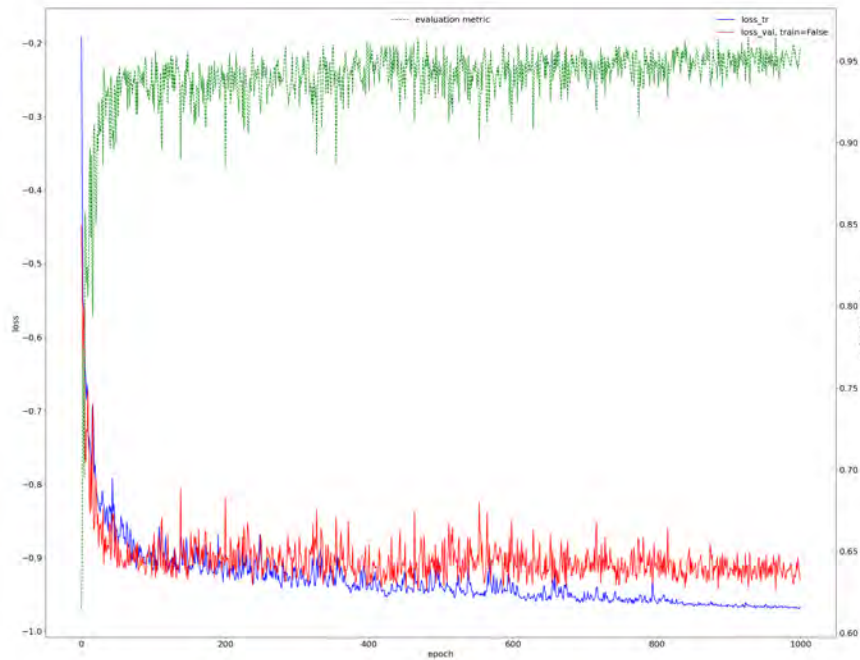
Figure 17: Per-epoch progress of Dice score coefficient, training loss and validation loss for the 4th training fold - RUMC train-val

compared to predictions coming from expert-level annotated training cases.

4.3. Qualitative Assessment of the Predictive Model for Vertebra Segmentation

The automatic predictions on RUMC inference set provides us with qualitative results. Covering a total of 25 output classes (24 vertebrae and the background class), the average runtime of a complete iteration over a single sample was a little over 110s, starting right from the initialisation step, normalizing the images read, resampling to isotropic resolution, moving further with traversing upwards along the spine for vertebral identification, labelling and final step of resampling predictions to original resolution. The number of visible vertebrae typically ranges from 7 to 19 completely visible vertebral levels. Even though in presence of a variety of scans in RUMC dataset, almost all of the observed anatomical landmarks potentially pose correct labels.

5. Discussion

5.1. Qualitative and Quantitative Estimations for Trained Detection Networks

Fig. 4 illustrates the possibility of including the missing aortic regions, which has an important effect postmodifications in the training scheme. We notice that the N_1 network trained on VUMC dataset only captures the key contrast details of the aorta, disregarding the presence of associated dilations. From the Tables 1 and 2, we deduce the capability of both networks to perform supervised learning in their limited spatial context paired with the training samples. The increase in voxel-level annotations improves the reliability of the trained network. However, attribution of this significant improvement goes to the inclusion of thrombus, aortic arc and ascending thoracic aorta for all patients in the training subset. Yet, the limited knowledge of vast range of surface walls in AA (upto 9 cm) and the possible generation of artifacts in the post-operative CT scans pose a potential chance to misfit the pipeline.

5.2. Qualitative Estimations for Vertebral Level Predictions

With regards to voxel-level performance of N_2 training scheme, we observe that Lessmann et al. (2019)'s prediction model produce realistic annotations and get the desired job done. We deduce that instance-based segmentation of vertebrae plays a crucial role in enhancing the chances of generalizing details of AAA, acting as an important marker for relative positioning with AA. The end-to-end pipeline makes best use of sharp resolution of the vertebrae, regardless of structural disparities in the anatomy of spine. When the qualitative results

Table 1: Segmentation metrics for the automated end-to-end learning compared with the expert annotations (VUMC dataset paired with annotations from the radiologists)

Metrics - Foreground Class	Mean Score
Average DSC of the ensemble	0.921
DSC in the best configuration	0.935
DSC of all the classes in the best configuration	0.903
Accuracy	0.999
False Discovery Rate	0.048
False Negative Rate	0.079
False Omission Rate	0.0001
False Positive Rate	0.000068473
Jaccard	0.879
Negative Prediction Value	0.999
Precision	0.952
Recall	0.921
Total Positive Reference	49222
Total Positive Test	47989
True Negative Rate	0.999

Table 2: Segmentation metrics for the automated end-to-end learning compared with the manual annotations (RUMC dataset paired with pseudo-labels updated by the author)

Metrics - Foreground Class	Mean Score
Average DSC of the ensemble	0.871
DSC in the best configuration	0.957
DSC of all the classes in the best configuration	0.945
Accuracy	0.999
False Discovery Rate	0.023
False Negative Rate	0.084
False Omission Rate	0.0002
False Positive Rate	0.000078414
Jaccard	0.896
Negative Prediction Value	0.999
Precision	0.977
Recall	0.916
Total Positive Reference	614999
Total Positive Test	590778
True Negative Rate	0.999
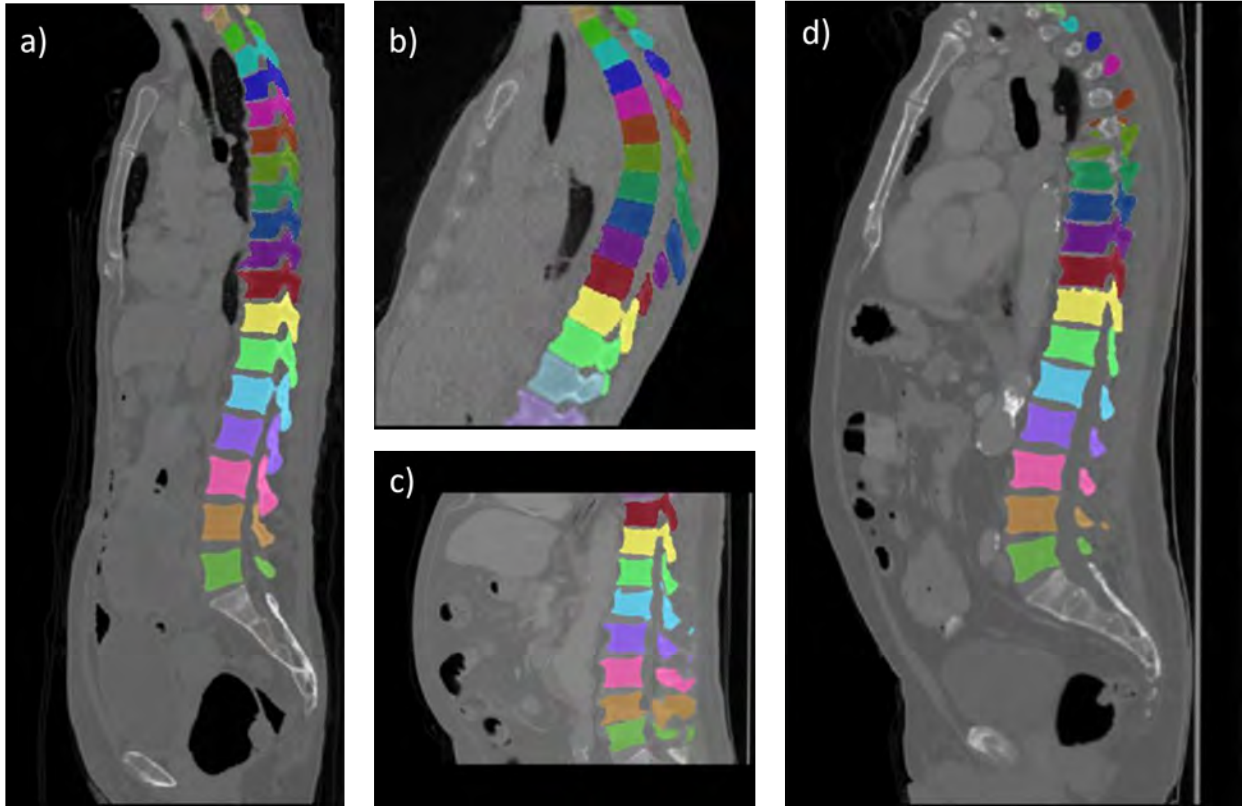
Figure 18: Qualitative assessment of the vertebrae segmentation - retreives the vertebra labels for a diverse set of CT scans *a)* female patient, *b)* thoracic cavity, *c)* abdominal cavity, *d)* typical AAA findings in male patients
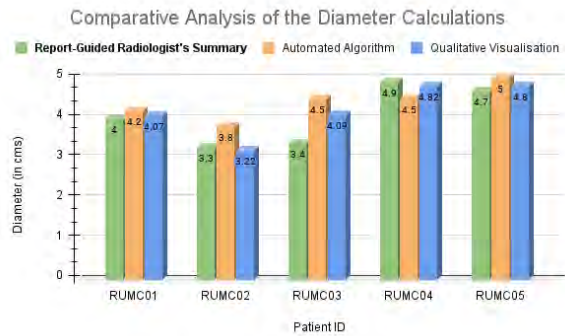
Figure 19: Comparative analysis of the calculated diameter across the detected aneurysm by expert-level feedback by the radiologists, qualitative conclusions by the author and the maximal diameter measured by the automated algorithm

are visually compared with the gold standard explanations illustrated by Ali Mirjalili et al. (2012) for our abdominal region of interest, the network excels to fit in our merging criterion with its equivalent predictions responsible for the essential characterization.

5.3. Comparative Analysis of Automated AAA Detection with Radiologists and Final Introspections

The multivariate pipeline results reflect a stark similarity while comparing the resultant measurements of the maximal diameter coming from end-to-end learning model, the approximate conclusions noted down by the radiologists speculating the enlargement with their field-expertise and finally a qualitative assessment on the original volumes by quantifying the respective maximum dilation by the author (Fig. 19). More specifically, P_{AAA} accounts for less than an average difference of 0.35 cm among a bunches of five randomly selected inference samples across the entire detection model. Although the close ties between the measurements, the proposed technique on occasions seems to overmeasure the cross-section. The possible reason could be the strange elliptical structures in the volumes resulting to a drift in calculations.

5.4. Limitations and Future Work

Analysing the impact of our trained N_1 networks, the raised question of dealing with the non-compliant cases expects a rather simple solution of adding the case-specific volumes paired with spatial information (for instance, the post-operative cases, the patients suffered from rAAA, very large aneurysm diameters, occasional sharp contrast of the lumen compared to CTA) in the training scheme to avoid the encountered failures. This increases the need of accurately annotating the enlargements. It is difficult to say the additional contribution of the metadata (i.e., age, gender, smoking and drinking,

associated diseases, etc.), although studies do discuss these risk factors but with limited reliability. We foresee to inculcate a first-hand strategy by getting expert annotations for our cohort and scale the quantitative improvements. A certain degree of confidence in the *Focal Loss* to (better) help with the struggles of class imbalance might make a difference. In future, the research looks up to perform the possible characterization of all the aneurysms across the entire aorta. To enable this wholesome structure, we will need to include the organs as landmark references for thoracic, thoracoabdominal and abdominal aortic aneurysm detection, on top of vertebral labels.

6. Conclusions

To summarize, the authors analyzed an end-to-end CAD system for automated voxel-level identification of the abdominal aortic aneurysms on CT images. We use a combination of an instance-based vertebra segmentation network and a post-processing pipeline to extract the characteristics of aneurysms in the abdominal aorta with excellent specificity using the unique automated nnU-Net setup. We demonstrate the brilliance of the self-configuring CNN architecture, which is based on a set of forth heuristic rules and their respective ways of pre-processing and post-processing patient-cases, demonstrating that it is a powerful tool for dealing with domain diversity and enhancing CNN performance with minimal effort towards the endless hassle of hyperparameter tuning.

The first component of supervised learning utilizing the VUMC dataset combined with detailed delineations was used to implement our proposed model on 398 RUMC cases. On the previous validation set, the training scheme achieves an average dice score coefficient of 93.5%. It moves further to synthesize the pseudo labels on the second dataset and carefully update the annotations to work along the problem statement. On the validation subsets, the CAD system exhibits a noteworthy average dice score coefficient of 95.7 % for the training of the RUMC dataset now paired with hand delineations on the anticipated labels. Given the expert radiologist and manual delineations, there appears to be a moderate agreement between the two training counterparts, suggesting a high capacity to generalize across the domain from a small number of training samples. The two training implementations' metrics are aligned, indicating that the flexible design decisions made automatically by the nnU-Net framework were successful.

To our understanding, the integration of prediction outputs in conjunction with anatomical surface landmarks to detect abdominal aortic aneurysms on CT scans is the first of its kind, trained only using updated pseudo-labels and textual guidances from the radiology reports for a comparative study to confirm its legitimacy.

The groundbreaking findings of this study encourage researchers to conduct research on holistic CAD systems that can be deeply intertwined into the clinical workflows, channeling the need of clinical expert involvement and assisting in the early diagnosis of aneurysms before the fateful event of rupture.

References

Adam, C., Fabre, D., Mougin, J., Zins, M., Azarine, A., Ardon, R., d'Assignies, G., Haulon, S., 2021. Pre-surgical and post-surgical aortic aneurysm maximum diameter measurement: Full automation by artificial intelligence. European Journal of Vascular and Endovascular Surgery 62, 869–877. doi:https://doi.org/10.1016/j.ejvs.2021.07.013.

Ali Mirjalili, S., McFadden, S.L., Buckenham, T., Stringer, M.D., 2012. A reappraisal of adult abdominal surface anatomy. Clinical Anatomy 25, 844–850. doi:https://doi.org/10.1002/ca.22119.

Antiga, L., Ene-iordache, B., Remuzzi, A., 2003. Centerline computation and geometric analysis of branching tubular surfaces with application to blood vessel modeling.

Antiga, L., Piccinelli, M., Botti, L., Ene-Iordache, B., Remuzzi, A., Steinman, D.A., 2008. An image-based modeling framework for patient-specific computational hemodynamics. Medical &amp Biological Engineering &amp Computing 46. doi:10.1007/s11517-008-0420-1.

Bengtsson, H., Bergqvist, D., Sternby, N., 1992. Increasing prevalence of abdominal aortic aneurysms. a necropsy study. The European journal of surgery= Acta chirurgica 158, 19–23.

de Boor, C., 2001. Smoothing and Least-Squares Approximation. pp. 207–242. doi:10.1007/978-1-4612-6333-3_14.

Brutti, F., Fantazzini, A., Finotello, A., Müller, L.O., Auricchio, F., Pane, B., Spinella, G., Conti, M., 2022. Deep learning to automatically segment and analyze abdominal aortic aneurysm from computed tomography angiography. Cardiovascular Engineering and Technology doi:10.1007/s13239-021-00594-z.

Caradu, C., Spampinato, B., Vrancianu, A.M., Bérard, X., Ducasse, E., 2021. Fully automatic volume segmentation of infrarenal abdominal aortic aneurysm computed tomography images with deep learning approaches versus physician controlled manual segmentation. Journal of Vascular Surgery 74, 246–256.e6. doi:https://doi.org/10.1016/j.jvs.2020.11.036.

Chaikof, E.L., Brewster, D.C., Dalman, R.L., Makaroun, M.S., Illig, K.A., Sicard, G.A., Timaran, C.H., Upchurch, G.R., Veith, F.J., 2009. SVS practice guidelines for the care of patients with an abdominal aortic aneurysm: Executive summary. Journal of Vascular Surgery 50, 880–896. doi:10.1016/j.jvs.2009.07.001.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3d u-net: Learning dense volumetric segmentation from sparse annotation, in: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016, Springer International Publishing, Cham. pp. 424–432.

de Bruijne, M., van Ginneken, B., Viergever, M.A., Niessen, W.J., 2004. Interactive segmentation of abdominal aortic aneurysms in cta images. Medical Image Analysis 8, 127–138. doi:https://doi.org/10.1016/j.media.2004.01.001.

Dziubich, T., Białas, P., Znaniecki, Ł., Halman, J., Brzeziński, J., 2020. Abdominal aortic aneurysm segmentation from contrast-enhanced computed tomography angiography using deep convolutional networks, in: Bellatreche, L., Bieliková, M., Boussaïd, O., Catania, B., Darmont, J., Demidova, E., Duchateau, F., Hall, M., Merčun, T., Novikov, B., Papatheodorou, C., Risse, T., Romero, O., Sautot, L., Talens, G., Wrembel, R., Žumer, M. (Eds.), ADBIS, TPDL and EDA 2020 Common Workshops and Doctoral Consortium, Springer International Publishing, Cham. pp. 158–168.

Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.C., Pujol, S., Bauer, C., Jennings, D., Fennessy, F., Sonka, M., Buatti, J., Aylward, S., Miller, J.V., Pieper, S., Kikinis, R., 2012. 3d slicer as an image computing platform for the quantitative imaging network. Magnetic Resonance Imaging 30, 1323–1341. doi:https://doi.org/10.1016/j.mri.2012.05.001. quantitative Imaging in Cancer.

Fitzgibbon, A., Pilu, M., Fisher, R., 1996. Direct least squares fitting of ellipses, in: Proceedings of 13th International Conference on Pattern Recognition, pp. 253–257 vol.1. doi:10.1109/ICPR.1996.546029.

Fleming, C., Whitlock, E.P., Beil, T.L., Lederle, F.A., 2005. Screening for abdominal aortic aneurysm: A best-evidence systematic review for the u.s. preventive services task force. Annals of Internal Medicine 142, 203–211. doi:10.7326/0003-4819-142-3-200502010-00012. pMID: 15684209.

Gao, J., Jiang, Q., Zhou, B., Chen, D., 2019. Convolutional neural networks for computer-aided detection or diagnosis in medical image analysis: An overview. Mathematical Biosciences and Engineering 16, 6536–6561. doi:10.3934/mbe.2019326.

Golla, A.K., Tönnes, C., Russ, T., Bauer, D.F., Froelich, M.F., Diehl, S.J., Schoenberg, S.O., Keese, M., Schad, L.R., Zöllner, F.G., Rink, J.S., 2021. Automated screening for abdominal aortic aneurysm in CT scans under clinical conditions using deep learning. Diagnostics 11, 2131. doi:10.3390/diagnostics11112131.

Habijan, M., Galić, I., Leventić, H., Romić, K., Babin, D., 2020. Abdominal aortic aneurysm segmentation from ct images using modified 3d u-net with deep supervision, in: 2020 International Symposium ELMAR, pp. 123–128. doi:10.1109/ELMAR49956.2020.9219015.

Hansen, N.J., 2016. Computed tomographic angiography of the abdominal aorta. Radiologic Clinics of North America 54, 35–54. doi:https://doi.org/10.1016/j.rcl.2015.08.005. cT Angiography.

Hong, H.A., Sheikh, U.U., 2016. Automatic detection, segmentation and classification of abdominal aortic aneurysm using deep learning, in: 2016 IEEE 12th International Colloquium on Signal Processing Its Applications (CSPA), pp. 242–246. doi:10.1109/CSPA.2016.7515839.

Hwang, B., Kim, J., Lee, S., Kim, E., Kim, J., Jung, Y., Hwang, H., 2022. Automatic detection and segmentation of thrombi in abdominal aortic aneurysms using a mask region-based convolutional neural network with optimized loss functions. Sensors 22. doi:10.3390/s22103643.

Isensee, F., Jaeger, P.F., Kohl, S.A.A., Petersen, J., Maier-Hein, K.H., 2020a. nnU-net: a self-configuring method for deep learning-based biomedical image segmentation. Nature Methods 18, 203–211. doi:10.1038/s41592-020-01008-z.

Isensee, F., Jäger, P., Wasserthal, J., Zimmerer, D., Petersen, J., Kohl, S., Schock, J., Klein, A., Roß, T., Wirkert, S., Neher, P., Dinkelacker, S., Köhler, G., Maier-Hein, K., 2020b. batchgenerators - a python framework for data augmentation. doi:10.5281/zenodo.3632567.

Jiang, Z., Do, H.N., Choi, J., Lee, W., Baek, S., 2020. A deep learning approach to predict abdominal aortic aneurysm expansion using longitudinal data. Frontiers in Physics 7. doi:10.3389/fphy.2019.00235.

Kent, K.C., 2014. Abdominal aortic aneurysms. New England Journal of Medicine 371, 2101–2108. doi:10.1056/nejmcp1401430.

Kent, K.C., Zwolak, R.M., Egorova, N.N., Riles, T.S., Manganaro, A., Moskowitz, A.J., Gelijns, A.C., Greco, G., 2010. Analysis of risk factors for abdominal aortic aneurysm in a cohort of more than 3 million individuals. Journal of vascular surgery 52, 539–548.

Kroell, N., 2021. imea: A python package for extracting 2d and 3d shape measurements from images. Journal of Open Source Software 6, 3091. doi:10.21105/joss.03091.

Landman, B., Xu, Z., Igelsias, J.E., Styner, M., Langerak, T.R., Klein, A., 2015. Segmentation outside the cranial vault challenge. doi:10.7303/SYN3193805.

Lareyre, F., Chaudhuri, A., Flory, V., Augène, E., Adam, C., Carrier, M., Amrani, S., Chikande, J., Lê, C.D., Raffort, J., 2021. Automatic measurement of maximal diameter of abdominal aortic aneurysm on computed tomography angiography using artificial intelligence. Annals of Vascular Surgery doi:https://doi.org/10.1016/j.avsg.2021.12.008.

Lee, K., Johnson, R.K., Yin, Y., Wahle, A., Olszewski, M.E., Scholz, T.D., Sonka, M., 2010. Three-dimensional thrombus segmentation in abdominal aortic aneurysms using graph search based on a triangular mesh. Computers in Biology and Medicine 40, 271–278.

Lessmann, N., van Ginneken, B., de Jong, P.A., Išgum, I., 2019. Iterative fully convolutional neural networks for automatic vertebra segmentation and identification. Medical Image Analysis 53, 142–155. doi:https://doi.org/10.1016/j.media.2019.02.005.

Li, B., Eisenberg, N., Witheford, M., Lindsay, T.F., Forbes, T.L., Roche-Nagle, G., 2022. Sex Differences in Outcomes Following Ruptured Abdominal Aortic Aneurysm Repair. JAMA Network Open 5, e2211336–e2211336. doi:10.1001/jamanetworkopen.2022.11336.

Li, X., Wen, Z., Zhu, H., Guo, Z., Liu, Y., 2020. An improved algorithm for evaluation of the minimum circumscribed circle and maximum inscribed circle based on the local minimax radius. Review of Scientific Instruments 91, 035103. doi:10.1063/5.0002233.

Lorensen, W.E., Cline, H.E., 1987. Marching cubes: A high resolution 3d surface construction algorithm 21, 163–169. doi:10.1145/37402.37422.

Lu, J.T., Brooks, R., Hahn, S., Chen, J., Buch, V., Kotecha, G., Andriole, K.P., Ghoshhajra, B., Pinto, J., Vozila, P., Michalski, M., Tenenholtz, N.A., 2019. Deepaaa: Clinically applicable and generalizable detection of abdominal aortic aneurysm using deep learning, in: Shen, D., Liu, T., Peters, T.M., Staib, L.H., Essert, C., Zhou, S., Yap, P.T., Khan, A. (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Springer International Publishing, Cham. pp. 723–731.

López-Linares, K., Aranjuelo, N., Kabongo, L., Maclair, G., Lete, N., Ceresa, M., García-Familiar, A., Macía, I, González Ballester, M.A., 2018a. Fully automatic detection and segmentation of abdominal aortic thrombus in post-operative cta images using deep convolutional neural networks. Medical Image Analysis 46, 202–214.

López-Linares, K., Lete, N., Kabongo, L., Ceresa, M., Maclair, G., García-Familiar, A., Macía, I, González Ballester, M., 2018b. Comparison of regularization techniques for dcnn-based abdominal aortic aneurysm segmentation, in: 2018 IEEE 15th Interna-

tional Symposium on Biomedical Imaging (ISBI 2018), pp. 864–867. doi:10.1109/ISBI.2018.8363708.

McAuliffe, M.J., Lalonde, F.M., McGarry, D., Gandler, W., Csaky, K., Trus, B.L., 2001. Medical image processing, analysis amp; visualization in clinical research, IEEE Computer Society, USA. p. 381.

Merkel, D., 2014. Docker: lightweight linux containers for consistent development and deployment. Linux journal 2014, 2.

Mohammadi, S., Mohammadi, M., Dehlaghi, V., Ahmadi, A., 2019. Automatic segmentation, detection, and diagnosis of abdominal aortic aneurysm (AAA) using convolutional neural networks and hough circles algorithm. Cardiovascular Engineering and Technology 10, 490–499. doi:10.1007/s13239-019-00421-6.

Pleumeekers, H., Hoes, A., van der Does, E., van Urk, H., Hofman, A., de Jong, P., Grobbee, D., 1995. Aneurysms of the Abdominal Aorta in Older Adults: The Rotterdam Study. American Journal of Epidemiology 142, 1291–1299. doi:10.1093/oxfordjournals.aje.a117596.

Raffort, J., Adam, C., Carrier, M., Ballaith, A., Coscas, R., Jean-Baptiste, E., Hassen-Khodja, R., Chakfé, N., Lareyre, F., 2020. Artificial intelligence in abdominal aortic aneurysm. Journal of Vascular Surgery 72, 321–333.e1. doi:https://doi.org/10.1016/j.jvs.2019.12.026.

Renapurkar, R.D., Setser, R.M., O'Donnell, T.P., Egger, J., Lieber, M.L., Desai, M.Y., Stillman, A.E., Schoenhagen, P., Flamm, S.D., 2012. Aortic volume as an indicator of disease progression in patients with untreated infrarenal abdominal aneurysm. European Journal of Radiology 81, e87–e93. doi:10.1016/j.ejrad.2011.01.077.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham. pp. 234–241.

Sokol, J., Nguyen, P.K., 2022. Risk prediction for abdominal aortic aneurysm: One size does not necessarily fit all. Journal of Nuclear Cardiology doi:10.1007/s12350-021-02680-0.

Taubin, G., Zhang, T., Golub, G., 1996. Optimal surface smoothing as filter design, in: Buxton, B., Cipolla, R. (Eds.), Computer Vision — ECCV '96, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 283–292.

Teng, P.y., Bagci, A.M., Alperin, N., 2011. Automated prescription of an optimal imaging plane for measurement of cerebral blood flow by phase contrast magnetic resonance imaging. IEEE Transactions on Biomedical Engineering 58, 2566–2573. doi:10.1109/TBME.2011.2159383.

Ulyanov, D., Vedaldi, A., Lempitsky, V., 2017. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. doi:10.48550/ARXIV.1701.02096.

Wanhainen, A., Verzini, F., Van Herzeele, I., Allaire, E., Bown, M., Cohnert, T., Dick, F., van Herwaarden, J., Karkos, C., Koelemay, M., Kölbel, T., Loftus, I., Mani, K., Melissano, G., Powell, J., Szeberin, Z., ESVS Guidelines Committee, de Borst, G.J., Chakfe, N., Debus, S., Hinchliffe, R., Kakkos, S., Koncar, I., Kolh, P., Lindholt, J.S., de Vega, M., Vermassen, F., Document reviewers, Björck, M., Cheng, S., Dalman, R., Davidovic, L., Donas, K., Earnshaw, J., Eckstein, H.H., Golledge, J., Haulon, S., Mastracci, T., Naylor, R., Ricco, J.B., Verhagen, H., 2019. Editor's choice – european society for vascular surgery (esvs) 2019 clinical practice guidelines on the management of abdominal aorto-iliac artery aneurysms. European Journal of Vascular and Endovascular Surgery 57, 8–93. doi:https://doi.org/10.1016/j.ejvs.2018.09.020.

Wolf, I., Vetter, M., Wegner, I., Böttger, T., Nolden, M., Schöbinger, M., Hastenteufel, M., Kunert, T., Meinzer, H.P., 2005. The medical imaging interaction toolkit. Medical Image Analysis 9, 594–604. doi:https://doi.org/10.1016/j.media.2005.04.005. iTK.

Zohios, C., Kossioris, G., Papaharilaou, Y., 2012. Geometrical methods for level set based abdominal aortic aneurysm thrombus and outer wall 2d image segmentation. Computer Methods and Programs in Biomedicine 107, 202–217. doi:https://doi.org/10.1016/j.cmpb.2011.06.009.

Appendix A. Voxel-Based Centerline Extraction

The experimented pipeline initiates its processing by taking the binary segmentations of aorta as input, and applying the distance transformation to generate the distance fields, namely : single-seed (SS) seeded field and boundary-seed (BS) seeded field for each aorta (Teng et al., 2011). The SS field approximates the shortest path between aorta voxels and the aortic root, whereas the BS field approximates the shortest distance between aorta voxels and the aortic boundary surface. The method intrinsically expects manual intervention to mark the seed points on the nodes of the aorta for extracting the centerline. Instead, we induced our proposition to automate the schedule using the two-dimensional shape measurements detected for the first and last slices with predicted pixel presence (typically, the central position of the maximum inclosing circle in shape [x, y] suggested by Kroell (2021)) along the axial plane. The pixel selection has a fundamental flaw in this approach, as it tends to ignore the ascending thoracic aorta and aortic arc because the goal is to indicate the end nodes of aorta, namely the aortic annulus, not the aortic arc's curvature point. For a better visual representation of the centerline, the extracted skeleton of the aorta is smoothed using cubic smoothing spline (de Boor, 2001). This processing has been disregarded in the final implementation to adopt an efficient method.
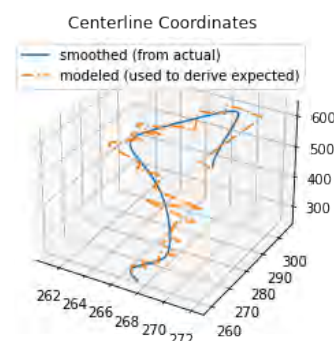
Figure A.20: Centerline extraction using distance transform
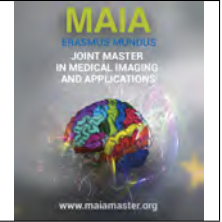
Mass Detection in Full Field Digital Mammograms with Multiscale Transformers

Amparo Soeli Betancourt Tarifa, Alessandro Bria

Università degli studi di Cassino e del Lazio Meridionale, Cassino, Italy

Abstract

Mammography is the mainstay imaging technique used for breast cancer screening and diagnosis, despite having recognized limitations it is applied worldwide to detect suspicious findings on breasts. Convolutional Neural Networks (CNNs) have been the de facto approach for automated medical image detection and diagnosis in the last decade. However, in recent years vision transformers have emerged as an alternative to CNNs. Particularly the Shifted Windows (Swin) transformer, a general purpose backbone capable of constructing hierarchical feature maps, yields interesting properties that could prove beneficial in medical imaging tasks.

In this work we investigate the potential of Swin Transformer as a backbone for mass detection in Full Field Digital Mammograms (FFDM). We mainly propose the use of Representative Points (RepPoints) and Deformable Detection Transformer (DETR) with Swin Transformer serving as a feature extractor for detecting masses on mammograms from the OPTIMAM Mammography Image Database (OMI-DB). The best transformer-based model obtained a True Positive Rate (TPR) of 0.903 at 0.8 False Positives per Image (FPpI) and an area under the Free Receiving Operating Characteristic (FROC) curve of 0.852 on FFDMs from Hologic scanners, outperforming previous work on OMI-DB and a competitive performance of 0.882 TPR at 0.8 FPpI and 0.812 area under the FROC curve, from our implemented baseline model.

Furthermore, we present an approach to combine predictions from the proposed models using Weighted Box Fusion (WBF), which achieves a meaningful improvement over single model performances. Finally, we propose applying this method to combine transformer and convolutional-based models predictions, further improving their performances. The TPR obtained by this last approach is 0.934 at 0.8 FPpI with an area under the FROC curve of 0.878.

Keywords: Mammography, Breast Cancer, Convolutional Neural Networks, Transformers, Mass detection

1. Introduction

1.1. Breast Cancer

In 2020 female breast cancer (BC) was the most commonly diagnosed cancer, with an estimated 2.3 million new cases (11.7%) and 685,000 deaths globally. Worldwide, BC also represented the leading cause of cancer deaths among women. (Sung et al., 2021).

Approximately half of breast cancers develop in women who have no identifiable BC risk factor other than gender (female) and age (over 40 years) (World Health Organization, 2022). Nonetheless, it is known that certain factors increase the risk of BC, including hormonal, lifestyle, and environmental changes (DeSantis et al., 2015).

Breast cancer arises in the epithelium of the ducts or lobules in the glandular tissue of the breast. Initially, it is confined to the duct or lobule where in general doesn't cause symptoms and has minimal potential to spread. Over time, these cancers may progress and invade the surrounding breast tissue, then spread to the nearby lymph nodes or to other organs in the body (World Health Organization, 2022).

BC treatment can be highly effective, therefore, when detected and treated early, the chances of survival are very high. Treatment of BC often consists of a combination of surgical removal, radiation therapy and medication (hormonal therapy, chemotherapy and/or targeted biological therapy) to treat the microscopic cancer that has spread from the breast tumor through the blood. Such treatment, which can prevent cancer growth and

spread, can thereby save lives.

1.2. Imaging modalities

The development and improvement of imaging technologies has significant value for the early detection of BC. Imaging modalities for diagnosis and staging of BC include mammography, ultrasound (US) and Magnetic Resonance Imaging (MRI).

Mammography is the base imaging technique used, by most developed countries, for BC screening and diagnosis in women over the age of 40 years (CDC, 2022). Mammography is a process that utilizes low-energy X-rays, the standard screening consists of mediolateral oblique (MLO) and craniocaudal (CC) views of each breast and aims to detect suspicious findings. Studies have shown a mortality reduction of about 40% (Nickson et al. (2012), Sankatsing et al. (2017), Broeders et al. (2012)) after mammography screening. However, with an overall sensitivity and specificity of 54.5% and 85.5%, as mentioned in the work of Aristokli et al. (2022), it has limitations, especially in women with dense breasts where cancer could be hidden in mammography. Therefore, women at increased risk for breast cancer are recommended to undergo additional screening with breast MRI.

Ultrasound is not often used as a primary diagnostic tool, but rather as a tool to further study a mammographic anomaly, to identify whether a soft tissue mass is solid or cystic and to distinguish benign from malignant masses in patients who present clinical symptoms. It is also employed if the patient has a clinical complaint or palpable abnormalities despite a negative mammogram.

Preoperative staging, response assessment to neoadjuvant therapy, evaluation of patients with cancer of unknown primary and screening of high-risk patients are some of the clinical indications for MRI in breast imaging (Mann et al., 2015). Dynamic contrast-enhanced MRI (DCE-MRI) provides high-resolution breast morphology and enables the depiction of both physiologic and morphological changes by obtaining MRI images before, during, and following the injection of a contrast agent, usually gadolinium-based. During DCE-MRI, tumors demonstrate rapid, intense enhancement followed by a relatively rapid washout compared to normal tissue making DCE-MRI the most sensitive modality for breast cancer detection (Hodler et al., 2019).

Each imaging technique has its own set of limitations and benefits, therefore they should be used in combination to aid in breast cancer staging and treatment.

1.3. Masses

In mammography, a mass is a space-occupying lesion seen in two separate projections and identified by its shape and contour. According to the BI-RADS system (Breast Imaging Reporting and Data System) by the American College of Radiology (ACR), a mass is characterized by its shape, contour, density with respect to normal fibroglandular tissue, association with other anomalies and its evolution over time, which can be observed when past mammograms are available (D'Orsi et al., 2018). A mass can fall in a BI-RADS category from 0-6 depending on its characteristics, where category 2 represents typically benign masses such as circumscribed masses with macrocalcifications or masses of fatty or mixed density; and category 5 corresponds to malignant masses, which are usually spiculated masses (Berment et al., 2014).

As previously mentioned, most circumscribed masses are benign. Nevertheless, due to specific histological characteristics, certain malignant lesions or lesions with a risk of malignancy may appear in the mammography in this falsely reassuring form and in rare cases, certain benign lesions may appear in the form of spiculated masses (Berment et al., 2014).

1.4. Computer Aided Systems

Computer-Aided Detection (CADe) and Computer-Aided Diagnosis (CADx) are the two types of Computer Aided systems. In our context, CADe primarily assists in the detection and localization of masses or anomalies that are present in medical imaging, leaving interpretation to the radiologist. CADx, on the other hand, assigns a classification to the masses and assists the radiologist in making decisions about the anomalies (Hassan et al., 2022).

Although mammography has recognized limitations, it is still the mainstay of BC screening due to its simplicity, fast acquisition and cost-effectiveness, especially compared to its high sensitivity counterpart DCE-MRI. Researchers and clinicians have implemented multiple strategies to improve mammography's performance, including doublereading (review of mammograms by two specialists), yearly screenings, obtaining two views per breast and analyzing previous mammograms for comparison. However, since mass detection is primarily a manual and difficult process, mainly dependent on the experience of physicians, a significant proportion of breast masses can be missed due to multiple reasons including visual fatigue and loss of attention (Wang et al., 2014).

Studies on the efficiency of using CADe systems as second opinion systems reveal that they can benefit even experienced radiologists by increasing their sensitivity from 77% to 85% and beginner radiologists by raising their sensitivity from 62% to 86% (Balleyguier et al., 2005). Thus the importance of developing precise CADe systems capable of behaving as a second opinion to aid physicians and support their decisions about the detection of masses in mammograms.

With significant advancements in the development of deep learning technologies over the last ten years, CADe systems have been predominantly built using

Convolutional Neural Networks (CNNs). CNNs are powerful networks for analyzing images because of their capability of preserving the image's spatial features and have been the de facto approach to automated medical image diagnosis in the past decade. However, the tremendous success of transformer architectures in the Natural Language Processing (NLP) domain has led researchers to explore its adaption to the computer vision field where it has emerged as a viable alternative to CNNs, delivering state-of-the-art performances in numerous computer vision tasks such as image classification, object detection (Zhang et al., 2022) and semantic segmentation (Liu et al., 2021a) to name a few, this while also demonstrating a number of interesting features that could be useful for medical imaging tasks.

Therefore, after the inception of Vision Transformers (ViT) in the work of Dosovitskiy et al. (2020), the medical imaging community has witnessed an exponential growth in the number of transformer-based approaches as seen in Figure 1. Nevertheless, for the problem of medical image detection, transformer-based techniques are less common than for segmentation and classification, as mentioned in Shamshad et al. (2022), and are mainly based on the detection transformer (DETR) framework (Zhu et al., 2020).

1.5. Our work

The main goal of our study is to explore the use of transformers as a backbone architecture for mass detection in mammography, investigating that way its potential for use in CADe systems and comparing its performance to CNNs. Additionally, we propose combining the detection predictions of the developed transformer-based models, their convolutional counterparts and both, in order to enhance the models strengths and mitigate their weaknesses achieving an overall improved performance and the development of a robust, effective CADe system. In spite of the fact that the use of transformers in medical images has grown in the past year, to the best of our knowledge this would be the first study to use a fully transformer-based architecture as a backbone for detection.

The remaining of this paper is organized as follows. Section 2 summarizes the existing work on mass detection in mammography including the data base used in this study. Section 3 presents the methods and networks used in this project along with the dataset. The key results obtained in the most relevant experiments are exposed in Section 4, and Section 5 presents the respective discussion of the results. Finally, in Section 6 the conclusions are given.

2. State of the art

From the early 1990s, academic and business circles have set off a research to develop computer-aided detection and diagnosis technology that can act as a second opinion or helper for radiologists. This research began with the use of traditional computer vision techniques, based on conventional machine learning and image processing techniques.

Ke et al. (2010) proposed a detection system based on texture features. They used bilateral comparison to detect the mass and the center of region of interest (ROI), followed by the calculation of fractal dimension and two-dimensional entropy as the texture features. Lastly, the type of ROI was classified by Support Vector Machine (SVM) as mass or normal region. The method achieved a sensitivity of 85.11% at 1.44 false positives per image (FPpI), in a total of 106 mammograms. Patel et al. (2019) presented an effective approach to detect masses in breast using Modified Histogram based Adaptive Thresholding (MHAT) method, tested it on more than 100 mammograms obtaining a true positive rate (TPR) of 98.3% at 0.78 FPpI. Years later, in the work of Mughal et al. (2017), texture features were also used along with color features to detect and classify masses. Methods, such as region growing were also proposed as in Punitha et al. (2018). The work uses an optimized region growing technique where the initial seed points and thresholds are optimally generated using a swarm optimization technique called Dragon Fly Optimization (DFO). Features are then extracted from the detected masses and passed to a Feed-Forward Network for classification. The approach achieved sensitivity of 98.1% Specificity of 97.8%, using 300 images from the Digital Database for Screening Mammography (DDSM).

Recently, many promising deep-learning models employed in computer vision, such as CNNs, transfer learning and deep learning-based object detection models, have shown considerable improvements in the performance of CAD systems. Therefore, several techniques for CAD systems have been presented based on the use of deep learning models.

Ribli et al. (2018), used fast R-CNN on a subset of the INbreast database with lesions, to classify and detect the malignant and benign lesions. They obtained 0.90 TPR at 0.30 FPpI. We can also find the work of Cao et al. (2021), who proposed a novel model for mass detection along with a new data augmentation technique to overcome overfitting, based on local elastic deformation which enhanced the performance of their model; however, its calculation speed is slower compared to the traditional augmentation techniques. This approach uses an enhanced, anchor free version of RetinaNet named FSAF (Zhu et al., 2019) for mass detection. As a result, the model achieved an average of 0.495 false-positive rate (FPR) per image for the INBreast dataset, while for the DDSM dataset each image has 0.599 FPR. Aly et al. (2021) proposed an end-to-end CAD system based on You Only Look Once-V3 with k-means generated anchors, which is an improved version of the network proposed by Redmon et al. (2016).
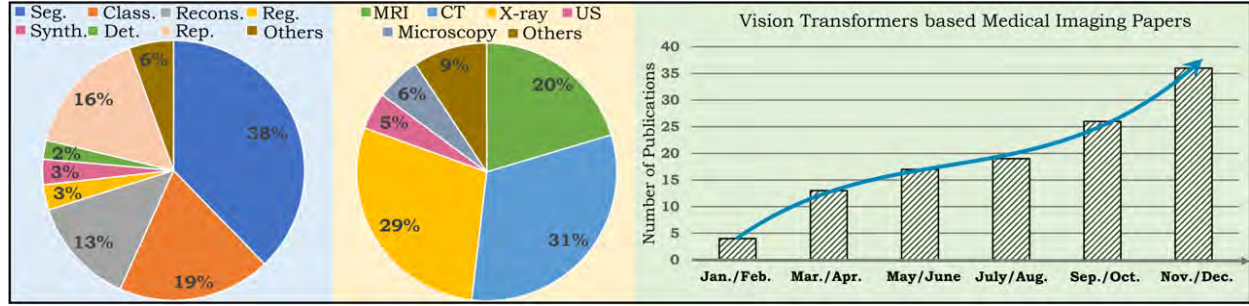
Figure 1: (Left) The pie-charts show statistics of the papers included in the survey presented by (Shamshad et al., 2022), according to medical imaging problem settings and data modalities. The rightmost figure shows consistent growth in the recent literature (for year 2021). Seg:segmentation, Class: classification, Recons: reconstruction, Reg: registration, Synth: synthesis, Det: detection, Rep: report generation, US: ultrasound.

2.1. Benchmark on OMI-DB

Agarwal et al. (2020), presented the benchmark of the performance of deep learning on the OPTIMAM Mammography Image Database (OMI-DB). In their work, a framework based on Faster R-CNN object detection model (Ren et al., 2015), using the whole FFDM (instead of patch-based strategy) for training and testing is proposed. A total of 7,245 images, obtained with Hologic scanners, originated from 2,042 positive cases with abnormalities and 842 normal cases, e.g. without any abnormalities, were used. The proposed framework achieved a True Positive Rate (TPR) of 0.87 at 0.84 FPpI on the test data.

3. Materials and methods

3.1. Dataset

OMI-DB is an extensive mammography image database composed of more than 2.5 million images from over 170,000 women, that were collected from three UK breast screening centres (Halling-Brown et al., 2020). It provides unprocessed and processed FFDMs, in DICOM format, from detected cancers along with normal and benign screening cases. The database also includes experts annotations and clinical data related to the images.

The database contains images from different scanner manufacturers such as Hologic Inc., Siemens, Philips and General Electric Medical Systems. For this study images from Hologic Inc. scanners were selected, as they represented the vast majority of images in the dataset. Since the focus of this work is detection of masses, our dataset referred to as OMI-H, consists of a total of 7,626 processed FFDMs from 1,945 patients, with both detected masses (positive images) and without abnormalities (negative images). Careful inspection of the overall selected images was performed, ensuring to discard images with artifacts or unwanted objects such as implants, marker clips or bands across the image. Furthermore, each case in the dataset may contain multiple images from the same patient.

3.2. Data preparation and pre-processing

The OMI-H dataset was divided into training, validation and test sets on patient basis to ensure that images from a particular case belonged exclusively to one of the three subsets. The division is performed, following the approach of Agarwal et al. (2020), on a 70-10-20 ratio for training, validation and test with a total of 1,361, 195 and 389 cases, respectively. Details on the number of images are provided in Table 1.

The mammograms were originally in DICOM format and were therefore converted to PNG (Portable Network Graphics) format for further use. The images on the dataset had pixel resolutions ranging from approximately 64 μm to 108 μm and sizes from 2,000 to 4,000 pixels.

In order to feed the network useful information only, the mammograms were cropped to contain the breast area of the image. This was done by applying triangle binarization to the original image followed by the extraction of the largest connected component yielding the mask of the breast, then the bounding box containing the mask was found and applied to crop the image as shown in Figure 2. Finally, due to computational limitations, the cropped images were downsampled to 200 μm pixel resolution.

It is worth mentioning that even though the amount of images and patients in our dataset is close to that of the work of Agarwal et al. (2020), it is not an exact match. Therefore, we can not ensure the same patients and images are part of our dataset.
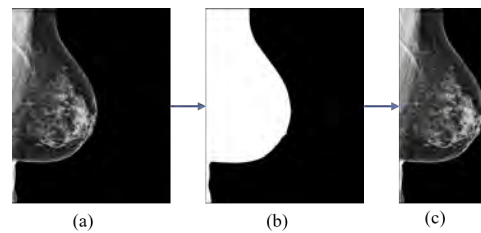
Figure 2: Breast area cropping : a) Original image b) Breast mask after breast-air segmentation c) Breast area cropped.

5.4

	Number of cases	Number of images	Positive Images	Negative Images
OMIDB-H	*1945*	*7626*	*3526*	*4100*
Train	1361	5339	2478	2861
Validation	195	766	349	417
Test	389	1521	699	822

Table 1: Description of dataset used in this work and its correspondent division.

3.3. Methodology

Our study is organized in three main stages. First, three object detection methods are trained using a novel, general purpose vision transformer backbone as feature extractor on its smallest variant. Second, we select two promising object detection methods and train them with a bigger variant of the backbone as well as with convolutional corresponding backbones for comparison. Finally, the predictions of the selected object detection methods, are combined into an ensemble to investigate whether convolutional-based and transformer-based detectors can complement each other and provide an overall boosted detection performance.

In addition to these steps, we train a baseline object detector to reproduce the work of Agarwal et al. (2020) since, as mentioned earlier, we can not ensure that previous work on OMI-DB Hologic mammograms works with the exact same images.

In the sections below, we describe the selected backbone, the object detection methods included in this study, the training process of our deep learning models, the bounding boxes fusion and the evaluation metrics used to assess the performance of the models.

3.3.1. Swin Transformer

Swin (**S**hifted **win**dow) transformer is a vision transformer capable of serving as a general purpose backbone for computer vision proposed by Liu et al. (2021b). Previously existing transformer-based models such as ViT directly conduct global self attention between all the fixed scale, non overlapping, medium-sized (16×16) image patches, which is unsuitable for high resolution images and dense tasks like image detection and segmentation, and is limited by its quadratic complexity. In contrast, Swin proposes a window shifting approach that limits the computation of self attention among small (4×4) patches within non overlapping local windows while also allowing cross window connection, thus achieving linear complexity to image size, making it suitable for dense vision tasks as well.

Overall Architecture

Firstly, the Swin Transformer architecture splits the input image into small non overlapping patches, of 4×4 pixels, by a *patch splitting module*. The raw pixel values of each patch are concatenated into feature vectors, of

dimension 4×4×3 = 48 and referred to as "tokens". The tokens are then passed to what are called stages 1, 2, 3 and 4 of the architecture.

- **Stage 1.-** At this stage, the tokens are projected to an arbitrary dimension "C" by a *linear embedding layer*. These tokens are passed to a pair of consecutive *Swin transformer blocks*. The first block processes tokens with a modified, shifted window based self attention, where attention is limited to a window that contains $M \times M$ neighboring patches and the second repeats the process, after displacing the window by ($[\frac{M}{2}]$, $[\frac{M}{2}]$), such that patches that were part of different windows and couldn't attend each other can now do so.

  On Figure 4 we present two successive Swin transformer blocks, where each consists of a shifted window based multi-self attention (MSA) module, followed by a 2-layer MLP with GELU activation in between. A normalization layer is applied before each MSA module and each MLP, also a residual connection is applied after each module.

- **Stage 2.-** Following Stage 1, tokens are merged by a *patch merging layer*, which concatenates tokens of 2×2 neighboring patches and forwards them to a linear layer that acts as a dimensionality reducer, where *4C*-dimensional concatenated tokens are downsampled to *2C* dimensionality. Several Swin Transformer Blocks are applied afterwards for feature transformation. This process is repeated in **Stage 3** and **Stage 4**, with different resolutions since tokens pass through the patch merging layer and with different number of Swin Transformer Blocks.

These stages together create a hierarchical representation with the same feature map resolutions as traditional CNNs such as Resnet (He et al., 2016). An overview of the Swin Transformer architecture is presented in Figure 3, which illustrates the base model.

Architecture Variants

The Swin base model, called Swin-B, was built to have similar size and computation complexity as ViT-B. Three variants of the base model were introduced: Swin tiny (Swin-T), Swin small (Swin-S) and Swin large (Swin-L) which have around 0.25×, 0.5× and 2× the complexity and size of Swin-B respectively. For our study, Swin-T and Swin-B have been used.

The default window size in all variants of Swin is set to *M=7* and the query dimension of each head is *d=32*. The architecture hyper parameters are presented on Table 2.

3.3.2. Object Detection Methods

In this study we used three object detection methods: RepPoints (Representative Points), Sparse R-CNN
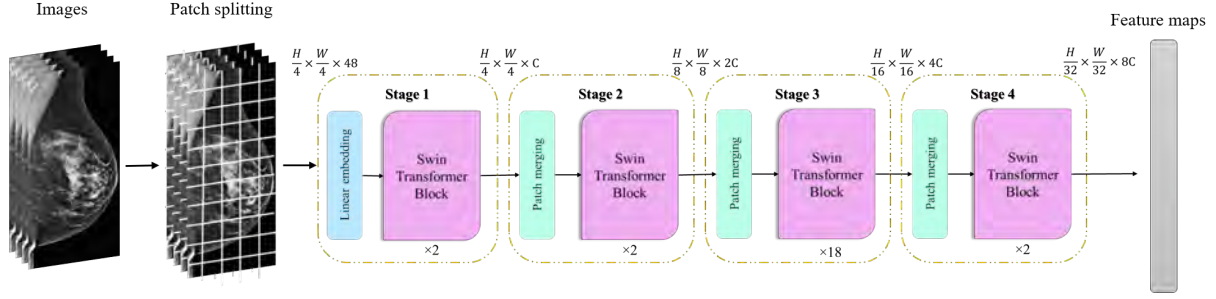
Figure 3: Architecture of Swin Transformer (Swin-Base)
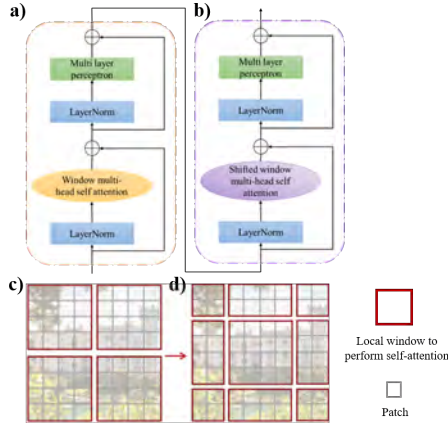
Figure 4: Two successive Swin Transformer Blocks : a) Swin transformer block with window self attention computation b) Swin transformer block with self attention computation on a shifted window. c) Ilustration of a window partitioning scheme where MSA is computed within each window. d) Ilustration of the resulting shifted windows where MSA is computed on the new windows.

Model	Embedding dimension "C"	Layer numbers
Swin-T	96	{2, 2, 6, 2}
Swin-S	96	{2, 2, 18, 2}
Swin-B	128	{2, 2, 18, 2}
Swin-L	192	{2, 2, 18, 2}

Table 2: Model variants architecture hyperparameters.

and Deformable Detection Transformer (DETR). The first two were used by the developers of Swin Transformer, to evaluate its performance on the COCO object detection challenge achieving promising results, therefore chosen by us, while deformable DETR is a novel transformer-based object detection model, which we believe could yield interesting results if combined with a transformer-based backbone. Additionally, RetinaNet was used but only with a convolutional backbone in order to have our own baseline by possibly replicating the performance of Agarwal et al. (2020) on our dataset.

RepPoints
RepPoints, illustrated in Figure 5, is an anchor free object detector which proposes a representation of ob-

jects as a set of sample points, suitable for both localization and recognition (Yang et al., 2019). The representative points learn to automatically organize themselves in a manner that bounds the spatial extent of an object and highlights semantically meaningful local areas when ground truth localization and recognition targets are given for training.

The training of RepPoints is driven jointly by object localization and recognition targets, such that the RepPoints are tightly bound by the ground-truth bounding box and guide the detector toward correct object classification.
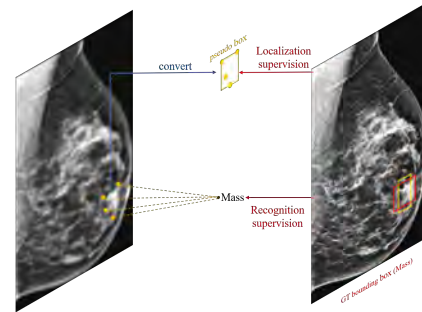
Figure 5: Reppoints

Sparse R-CNN
The work of Sun et al. (2021) presents Sparse R-CNN, a purely sparse method for object detection in images where a predetermined sparse set of learned object proposals are fed to an object recognition head for classification and location. As shown in Figure 6, a fixed small set of learnable bounding boxes, represented by 4-d coordinates, are given to object candidates which are used as proposal boxes to extract the ROI feature by ROIAlign.

The learnable *proposal boxes* are the statistics of possible object location while the 4-d coordinate is a rough object representation. Another important concept introduced on this work is *proposal feature*, which is a high-dimension (e.g., 256) latent vector expected to encode the rich instance attributes better than the rough bounding box. Proposal feature, in particular, generates a set of tailored parameters for its unique object

recognition head. *Proposal boxes* and *proposal features* are both randomly initialized and optimized, along with other network's parameters.
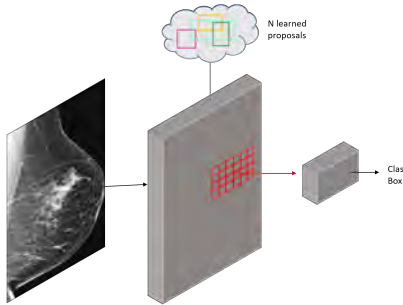
Figure 6: Sparse R-CNN

Deformable DETR

Carion et al. (2020) proposed DETR, an end-to-end object detection framework mainly characterized by the use of a set-based global loss which enforces unique predictions via bipartite matching and a transformer encoder-decoder architecture. While DETR removes the need of hand designed components such as anchor generation, which directly encodes our prior knowledge of the task, it also suffers from limited feature spatial resolution and slow convergence.

Deformable DETR aims to mitigate DETR's issues by combining the best of deformable convolution's sparse spatial sampling and Transformers' relation modeling capability. It also proposes a *deformable attention module*, which attends to a restricted number of sample locations as a pre-filter for significant key components out of all the feature map pixels, the module can be extended to aggregate multi-scale features. Deformable DETR replaces transformer attention modules processing feature maps by multi-scale deformable attention modules.

3.3.3. Network Training

Our first and main proposed task was the training of different object detection methods using Swin Transformer as a feature extraction backbone. Our work focuses on the use of Swin-T and Swin-B as backbones to assess also the impact of a smaller and deeper transformer backbone respectively. In addition, the detection frameworks were trained with convolutional backbones for further analysis and comparison, the chosen CNNs were Resnet 50 and Resnet 101.

In this study we use MMDetection (v.2.24.1), a PyTorch-based open source object detection toolset, presented by Chen et al. (2019), to train all the proposed frameworks. It is worth mentioning that, Liu et al. (2021b) conducted their experiments on COCO 2017 for object detection and instance segmentation using this toolset and therefore it was also selected by us to run our experiments.

Pre- processing pipeline

Our dataset sub division was closely balanced between mammograms with masses and negative mammograms among training, validation and testing sets. The image pre-processing pipeline consists, for all three frameworks, of replicating the single channel information into 3 channels, multi-scale training achieved by resizing the input images such that the shorter side is between 480 and 800 while the longer side is at most 1333 pixels and setting to true the *'keep aspect ratio'* parameter, followed by the normalization to the default mean and standard deviation used in the pre-trained setup. Due to memory and computational limitations, in the case of Deformable DETR the images were resized between 362 to 600 on the shorter side and 1000 pixels at most on the longer side.

Data augmentation

Although Swin doesn't require large-scale training datasets (i.e., JFT-300M) as ViT to achieve high performances, to the best of our knowledge it could benefit from more data. Therefore several data augmentation policies were added to the pipeline using MMDetection's *AutoAugment* class, which is an implementation of the data augmentation strategies proposed by Zoph et al. (2020).

AutoAugment is provided with a list of "policies" where each component is a specific augmentation policy, and can be composed by several augmentations and a probability of being applied. When AutoAugment is called, a random policy in "policies" will be selected and applied to augment images with a certain probability, if given. The list of policies used in our study consists of:

- Horizontal flip, applied with a probability of *p=0.5*.

- Random crop.

- Contrast transformation, with magnitude values of [0.4,0.8,1.5] and a probability of *p=0.5*.

- Brightness transformation, with magnitude values of [0.3,0.7,1.3] and a probability of *p=0.5*.

For deeper backbones (Swin-B and Resnet 101), the probabilities are increased to *p=0.6* and for training RetinaNet the only augmentation applied was horizontal flip as in the reference work of Agarwal et al. (2020).

Training

MMDetection provides a collection of pre-trained detection models, which includes models pre-trained on MSCOCO dataset (Lin et al., 2014). Pre-trained models of our selected object detection methods with a Swin Transformer backbone were not available, thus for training we use the pre-trained Imagenet weights of Swin,

Method	Backbone	Learning rate	Optimizer	Epochs
RepPoints	Swin-T	1.25e-05	AdamW	19
	Swin-B	1.25e-05	AdamW	32
	Resnet50	1.00e-04	SGD	22
	Resnet101	1.00e-04	SGD	16
Deformable DETR	Swin-T	1.25e-05	AdamW	28
	Swin-B	1.25e-05	AdamW	18
	Resnet50	1.25e-05	AdamW	24
	Resnet101	1.25e-05	AdamW	24
RetinaNet	Resnet50	7.81e-05	SGD	13
Sparse R-CNN	Swin-T	3.13e-06	AdamW	23

Table 3: Training parameters

Model	Backbone	Weight
RepPoints	Swin-T	2.0
	Swin-B	1.3
	Resnet50	1.7
	Resnet101	1.3
Deformable DETR	Swin-T	1,7
	Swin-B	0.4
	Resnet50	0.4
	Resnet101	0.1

Table 4: Weights assigned to models after applying grid search.

Ground truth and predictions   After NMS   After WBF

Figure 7: Schematic illustration of NMS and WBF outcomes for an ensemble of inaccurate predictions (Blue – different models' predictions, yellow – ground truth).

provided by the authors, along with our object detection methods pre-trained with CNNs for fine-tuning. Pre-trained models of the object detection methods were available with convolutional backbones and were therefore used to fine-tune the convolutional models.

Training of all models was done using two 16 GB, NVIDIA Tesla V100 GPUs, with a batch size of two distributed across them. During fine-tuning the epoch with the highest mean Average Precision (mAP) over IoU thresholds from 0.1 to 0.5 (step 0.05) was saved and selected as the best model, this metric was also used to monitor the training of the models, and early stop if necessary. All models converged before epoch 36. Table 3 presents additional settings used in the fine-tuning of the models. For RetinaNet we follow the recommendation in Agarwal et al. (2020) for the anchor boxes scales.

3.3.4. Weighted Boxes Fusion

Weighted boxes fusion (WBF) is a method for merging predictions from several object detection models, proposed by Solovyev et al. (2021). Ensemble methods have been widely used in machine learning, since combining predictions from different models usually yields more accurate results than a single model and has the potential of better generalization .

Unlike Non-Maximum Suppression (NMS) and soft-NMS methods that discard part of the predictions, WBF uses the confidence scores of all proposed bounding boxes to generate the averaged boxes as shown in Figure 7.

In this study we use the authors implementation, available on their GitHub repository. To give as input to the WBF method, we produce a list of predictions from the trained models, a second list with their respective scores and a third one with the labels of the predictions. Since our models have single class predictions (Mass), the labels list is an array of ones of the same length as the previously mentioned lists.

Each model has to be given a weight, to define such weights we applied a grid search on weights between 0.1 and 2 with a step of 0.3, for each model using their predictions on the validation data. The weights that achieve the highest area under the FROC curve, with FPpI computed on negative images, are then used to perform WBF on the test data. The selected weights

are presented on Table 4. Another needed parameter is the IoU threshold, which is set to 0.05.

It is worth mentioning that the predictions fed to the WBF method are NMS processed outputs from the models, in the case of deformable DETR which doesn't have NMS as part of its post-processing pipeline we apply our own implementation of NMS to its predictions. Authors of WBF previously experimented applying their method on raw model outputs without NMS and concluded that WBF works well for combining boxes of relatively accurate models, however, it performs worse than NMS when there are a large number of overlapping boxes with varying confidence levels.

WBF is applied on a selection of models with Swin backbone only, models with convolutional backbones only and finally a combination of both. This is done to see whether this method might help boost models' individual performance and foremost assess if combining convolutional and transformer-based models can enhance their strengths while mitigating their weaknesses.

3.4. Evaluation Metrics

The True Positive Rate (TPR), also known as sensitivity or recall, is a widely used metric to evaluate the performance of CADe systems in breast mass detection. TPR is commonly reported in a range of 0.75 to 0.85 FPpI in commercially available CADe systems , thus we provide the TPR on different FPpI values (0.75, 0.8, 0.85).

To assess and compare approaches, the area under the curve (AUC) of the Free-response Receiver Operating Characteristic (FROC) curve is also employed. As an output of the network the confidence score of the predicted bounding boxes is obtained and used to plot the FROC by considering the bounding boxes above a given threshold, which is increased from 0 to 1 in intervals of

0.05. To calculate the AUC we consider the TPR in a range of FPpI ∈ [0,1].

The TPR is computed using equation 1, where TP and FN stand for true positives and false negatives per image. A prediction bounding box is considered a true positive when its Intersection Over the Union (IoU) with the ground truth is equal or greater than 10 %, this value was chosen following the recommendations of Agarwal et al. (2020). Since our dataset contains certain cases with multiple masses, the IoU is computed for each ground truth and follows the same criteria to determine if they are TPs, if they remain undetected they are considered FNs and the predicted bounding boxes which don't have a IoU greater than the assigned threshold are considered FPs. In the case of normal images, all predictions are considered FPs.

$$TPR = \frac{TP}{TP + FN} \qquad (1)$$

The FPpI is calculated by computing the amount of FPs found in the images and dividing it by the total of images. In the benchmark work, the FPpI is determined using both positive and negative images, in order to compare our baseline model we follow the same evaluation approach. Nevertheless, after establishing the baseline model we compute the FPpI in negative images only, to assess how our models perform in mammograms without masses.

4. Results

In this section, the performances of the trained mass detection models are presented in four separate sections, firstly we present our baseline, the second and third sections present results according to the models' feature extraction backbone and the fourth which presents the performance of WBF applied to different models.

4.1. Baseline

Our first experiment has the goal of serving as a baseline for our dataset, for this purpose we trained a RetinaNet with a Resnet50 backbone. It can be seen in Figure 8 that our model's FROC curve closely approximates the FROC curve presented by Agarwal et al. (2020) on the test data, furthermore authors reported achieving a TPR=0.87 at 0.84 FPpI and our model reaches a TPR=0.88 at 0.84 FPpI. The FPpI is computed on both positive and negative images for fair comparison, then we also recalculated the FROC with FPpI on negative images only, to be consistent with the rest of the experiments.

4.2. Object detection methods and Swin transformer

4.2.1. RepPoints, Deformable DETR and Sparse-RCNN with Swin-T

As the first attempt of this set of experiments, RepPoints, Deformable DETR and Sparse R-CNN are

(a)

(b)

(c)
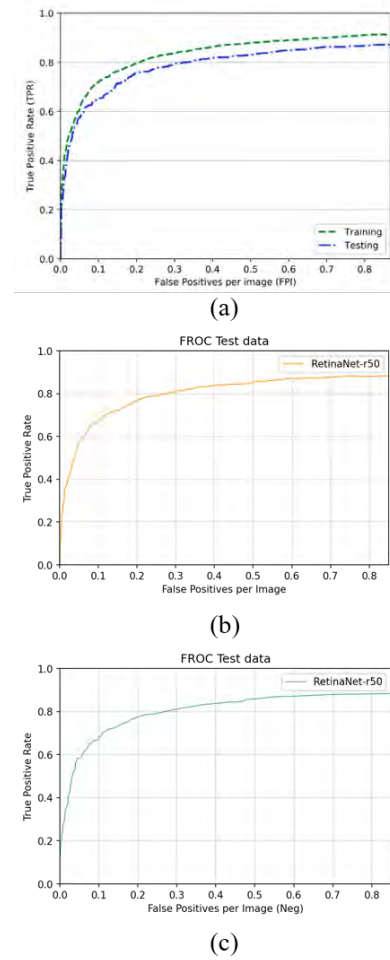
Figure 8: Free-response Receiver Operating Characteristic (FROC) curves of (a) Agarwal et al. (2020) (b) RetinaNet-Resnet50 (FPpI computed on all images) and (c) RetinaNet-Resnet50 (FPpI computed on negative images) on test data.

trained with a Swin-T backbone. Their performance is evaluated using the FROC on validation and test data as well as their respective TPRs at 0.75,0.8 and 0.85 FPpI.

Table 5 presents the TPR of the models at the previously mentioned FPpI values, on validation and test data. The FPpI is calculated using negative images only.

RepPoints and deformable DETR are selected for further experiments as their performance at the FPpI range of interest is consistently better on both validation and test data. Figure 10 presents the plotted FROC curves of these two models, where it can be observed that they already outperform our baseline model, with a TPR=0.908 at 0.85 FPpI (RepPoints-Swin-T) compared to a TPR=0.884 (RetinaNet-Resnet50).

Table 6 contains the area under the FROC curve achieved by these models, in which it can be seen once again that RepPoints-Swin-T surpasses the baseline model's performance. In Figure 9 examples of mass detection results given by these 3 models are shown.

Model	Backbone	Validation data			Test data		
		TPR at 0.75	TPR at 0.8	TPR at 0.85	TPR at 0.75	TPR at 0.8	TPR at 0.85
RepPoints	Swin-T	0,887	0.893	0.896	0.899	0.903	0.908
	Swin-B	0.875	0.878	0.886	**0.901**	0.903	0.905
	Resnet50	**0.893**	**0.896**	**0.900**	**0.901**	**0.906**	**0.910**
	Resnet101	0.887	0.892	0.896	0.898	0.902	0.907
Deformable DETR	Swin-T	0.875	0.875	0.879	0.872	0.876	0.880
	Swin-B	0.867	0.868	0.868	0.883	0.886	0.889
	Resnet50	0.810	0.811	0.812	0.825	0.825	0.825
	Resnet101	0.837	0.839	0.839	0.862	0.864	0.865
Sparse R-CNN	Swin-T	0.866	0.868	0.869	0.867	0.869	0.869
RetinaNet	Resnet50	-	-	-	0.881	0.882	0.884

Table 5: True positive rate values on different FPpI values. (FPpI calculated on negative images)

Model	Backbone	Validation data AUC FROC	Test data AUC FROC
RepPoints	Swin-T	0.830	0.847
	Swin-B	**0.832**	**0.852**
	Resnet50	0.829	0.842
	Resnet101	0.8258	0.844
Deformable DETR	Swin-T	0.816	0.805
	Swin-B	0.811	0.825
	Resnet50	0.771	0.781
	Resnet101	0.781	0.808
RetinaNet	Resnet50	-	0.812

Table 6: Area under FROC curves of trained models.
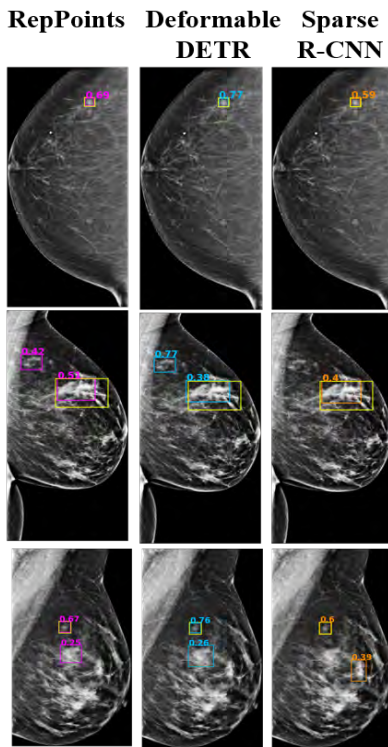
Figure 9: Mass detection results obtained by RepPoints, Deformable DETR and Sparse R-CNN with Swin transformer backbone.(yellow: GT box, purple,blue and orange: each models respective prediction boxes.The numbers shown correspond to the confidence score)

4.2.2. RepPoints and Deformable DETR with Swin-B

The following experiments use the previously selected object detection methods with a Swin-B back-
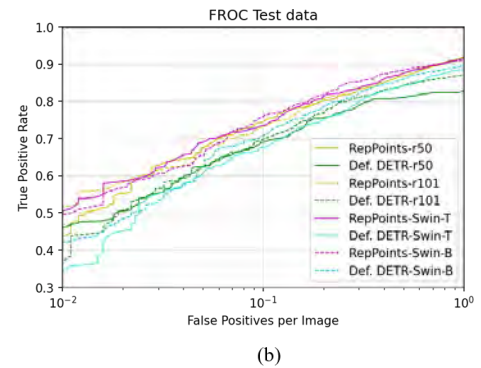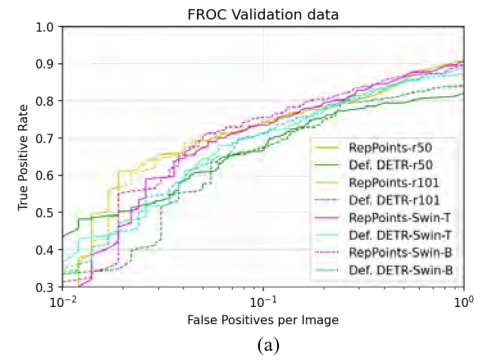
Figure 10: FROC curves of RepPoints and Deformable DETR with different backbones (Swin-T, Swin-B, Resnet 50 and Resnet 101) on (a) Validation and (b) Test data.

bone, with a higher probability on the data augmentation policies. The results obtained by these models are presented on Table 5 where it can be observed that the TPR values on 0.75-0.85 FPpI are very similar to those achieved by the same methods with Swin-T backbone. Nevertheless, the area under the FROC curves of the Swin-B versions are slightly higher as shown in Table 6. The plotted FROC curves can be observed on Figure 10.

4.3. RepPoints and Deformable DETR with CNNs

As mentioned in section 4.2.1, RepPoints and Deformable DETR were chosen as the object detection methods to use for further experiments. To have equivalent convolutional-based models for comparison, we train the detection heads with Resnet 50 and Resnet 101
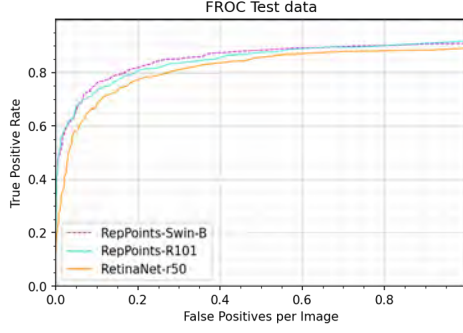
Figure 11: Comparison of the FROC curves of the best performing transformer and convolutional-based models with the baseline model.

Model	Backbone	TPR at 0.75	Test data TPR at 0.8	TPR at 0.85
WBF (RepPoints+ Deformable DETR)	Swin-T+Swin-B	0.922	0.926	0.929
	Resnet50+Resnet101	0.920	0.921	0.922
	Swin-T+Swin-B+ Resnet50+Resnet101	**0.933**	**0.934**	**0.936**

Table 7: TPR at different FPpI values (computed on negative images) of WBF of different models.

which are comparable in terms of parameter counts, to Swin-T and Swin-B.

Results obtained by these models are shown in Tables 5 and 6. The area under the FROC curves of Rep-Points with both convolutional backbones as well as the TPR (0.75-0.85 FPpI) are almost equal to those of their transformer counterparts. However, Deformable DETR with the convolutional backbones underperforms when compared to the transformer-based model.This can also be appreciated on the plotting of their respective FROC curves presented on Figure 10.

Figure 11 shows the FROC curves of the best transformer (RepPoints-Swin-B) and convolutional (RepPoints-Resnet 101) based models along with our baseline. The superiority and similarity of both models can be appreciated through all FPpI values of the curve.

4.4. Weighted boxes fusion

In order to improve our detection predictions, WBF is applied combining the outputs of all transformer-based models, all convolutional models and finally both. The positive impact of WBF can be qualitatively appreciated on the examples presented in Figure 12, where it gives more accurate prediction coordinates and also discards wrong predictions. Figure 13 presents the plotting of the FROC curves of the all the WBF models along with the best performing transformer and convolutional-based models and our baseline for qualitative comparison.

4.4.1. WBF applied on transformer-based models

Firstly the predictions of RepPoints and Deformable DETR with Swin transformer backbones are combined.
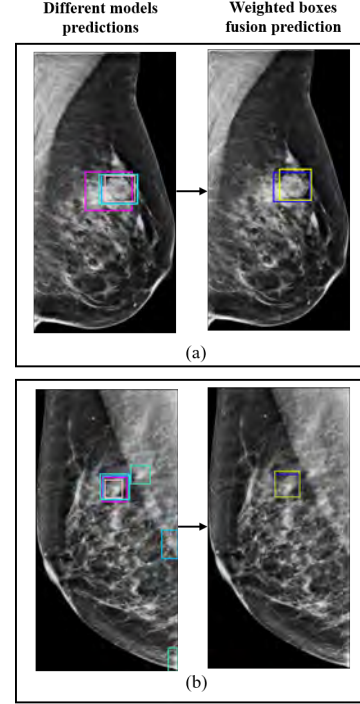
Figure 12: Weighted boxes fusion results from different models predictions (a) Gives more accurate coordinates of the prediction box (b) Discards wrong predictions and outputs an accurate prediction. (yellow: GT box, blue: WBF prediction, color boxes:different models predictions)

Model	Backbone	Validation data AUC FROC	Test data AUC FROC
WBF (RepPoints+ Deformable DETR)	Swin-T+Swin-B	0,859	0,869
	Resnet50+Resnet101	0,843	0,863
	Swin-T+Swin-B+ Resnet50+Resnet101	**0,863**	**0,878**

Table 8: Area under the FROC curves obtained by WBF models. (FPpI calculated on negative images)

Table 7 presents the TPR values obtained, in which it can be seen there's a 2% improvement on the TPR at 0.85 FPpI compared to the highest value reached individually by a transformer-based model (RepPoints-Swin-T). The same improvement is reflected on the area under the FROC curve of this model, shown in Table 8.

4.4.2. WBF applied on convolutional-based models

Results obtained by the fusion of convolutional-based models are presented on Tables 5 and 8. As in the case of combined transformer-based models, there is an improvement compared to the individual perfomances of convolutional-based models on a similar proportion.

However, when comparing the results of WBF of convolutional vs. transformer-based models, the last obtain slightly better results in terms of TPR and area under the FROC curve.
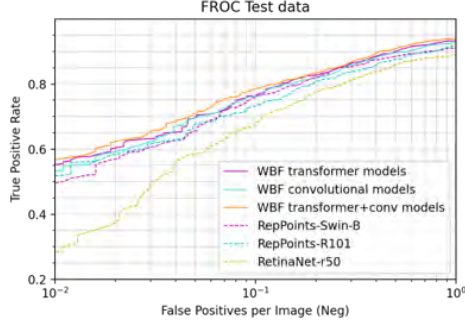
Figure 13: Comparison of FROC curves of baseline model, best performing convolutional and transformer-based models to WBF applied to transformer-based, convolutional-based and all models.

			Test data AUC FROC
Model	**Backbone**	**IoU**	**(FPpI neg)**
WBF (RepPoints+ Deformable DETR)	Swin-T+Swin-B+ Resnet50+Resnet101	0.1	0.878
		0.2	0.873
		0.25	0.869
		0.3	0.864

Table 9: Area under the FROC curves obtained by applying WBF to all models predictions and varying the IoU threshold for evaluation.

4.4.3. WBF applied on all models

After observing a significant improvement by combining models of the same type of backbone, they are all combined to see if further improvement is possible. As shown in Tables 5 and 8, combining all models improves the results of WBF applied transformer or convolutional- based models alone and when compared to the performance of individual models we can observe a 2.5% improvement.

On Figure 14 examples of mass detection results obtained by this combination are presented.

Additionally we carried out some experiments to assess the performance of WBF of all models, while modifying the IoU threshold. Previously all trials were evaluated on an IoU=0.1 to define TPs and FPs, on this set of experiments we increase the threshold to 0.2, 0.25 and 0.3. The obtained FROC curves for these trials are plotted in Figure 15, it can be observed these models still outperform the baseline and individual performance of RepPoints-Swin-B. Obtaining a TPR of 0.93, 0.92 and 0.92 at 0.85 FPpI, for a 0.2, 0.25 and 0.3 IoU respectively.

Table 9 presents the area under the FROC curves of these experiments, with FPpI calculated on both negative images and all images obtaining very similar values.

5. Discussion

This study focuses on the detection of masses on Full Field Digital Mammograms using transformer-based ar-
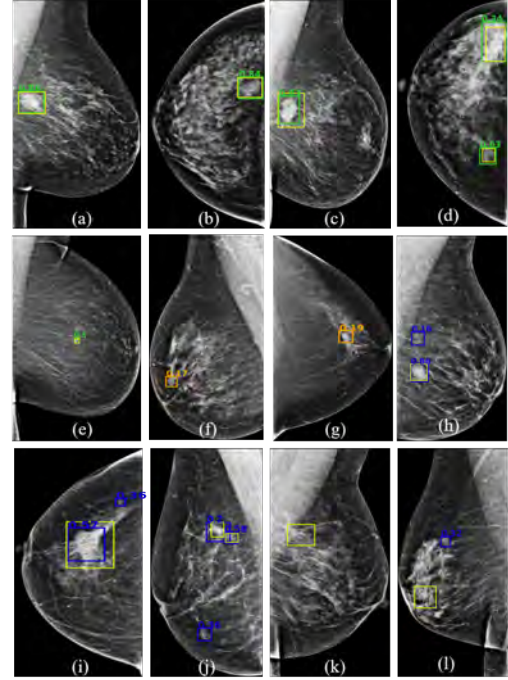
Figure 14: WBF of all models: Mass detection results on test data , (a–e) present TP detections, (f, g) show some FP detections on negative images, (h-j) show TP and FP detections on positive images, and (k, l) show undetected masses (yellow: GT box, green, orange and blue: detection boxes). The numbers shown correspond to the confidence scores of the predictions.
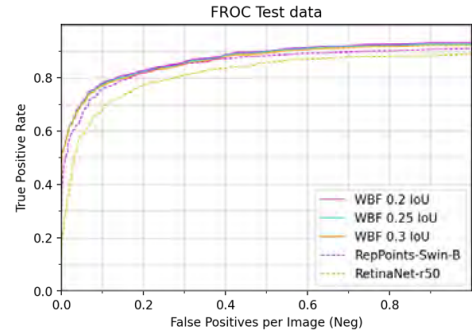
Figure 15: Comparison of FROC curves of the baseline model, RepPoints-Swin-B, and WBF of all models evaluated at different IoU thresholds.

chitectures. To our knowledge the proposed work is the first to attempt implementing the use of a transformer backbone for mass detection in mammograms, resulting in models that outperform previous state of the art methods.

Experiments have shown that achieving good results on mass detection using transformer backbones is possible and promising. The results of this work show on par performances using transformer and convolutional backbones for object detection methods, even when convolutional models benefit from pre-trained heads on CNNs. Based on the achieved results, we believe transformers have the potential to outperform CNNs, on

this task, considering that the Swin Transformer backbone was used unmodified, as proposed by its authors. Exploring the modification of Swin's hyperparameters, such as *window size*, could lead to extracting better information for mass detection. Additionally, an improved second version of Swin, capable of handling higher resolution images, has been released and its use could lead to improved performances (Liu et al., 2021a).

An interesting result seen on the experiments is the performance of Deformable DETR with a transformer backbone in contrast to its convolutional peer. While RepPoints performs similarly with both backbones, it seems that the transformer head Deformable DETR specially benefits from transformer extracted features. This finding may encourage researchers to utilize this object detection method with a transformer backbone instead of a CNN for better results. It is worth mentioning that Deformable DETR achieves a competitive performance although it has been trained with a smaller image size, which opens the possibility of not only achieving on par results to RepPoints if possible to train it with a higher resolution, but also highlights the potential of this method combined with a transformer backbone.

Additionally the use of weighted boxes fusion has been proven useful to boost the performance of object detection models. As shown, combining only transformer-based models already achieves a meaningful improvement over the baseline model and even outperforms the ensemble of convolutional-based models in respect to the area under the FROC curve and sensitivity. Moreover, further improvement can be achieved when both convolutional and transformer-based models are ensembled, which suggests these backbones may extract different features that complement each other when combined together. Due to time limitations the grid search for the optimal weights was performed on a small set of seven options, this search could be expanded to possibly find better weights for the ensembles. It also can be observed that the best weights found, assign a low weight to Deformable DETR models with convolutional backbones while higher weights are assigned to its transformer peers, suggesting these models have a more positive impact when performing the fusion of the predictions.

Finally, our last experiments showed that the predictions of WBF, of convolutional and transformer-based models, can also achieve promising results with higher IoU thresholds for evaluation. Since it can be said a IoU= 0.1 is small, considering computer vision projects use a IoU=0.5 to compute FPs and TPs, we've shown that even with 3× this value it has been possible to obtain robust predictions in the medical image domain. After observing this performance, there is a possibility similar results can be achieved also combining transformer-based models only, which could be investigated in future work.

6. Conclusions

This study presents the implementation of RepPoints, Deformable DETR and Sparse R-CNN models, with the general purpose vision transformer backbone *Swin*, for mass detection in mammograms from a high resolution, large scale dataset. It was shown that transformer-based models can be fine-tuned on pre-trained on natural images models and be successfully adapted to detect masses in mammograms. The implemented models achieve promising results on this task and show on par, or even superior, performances to their convolutional counterparts. Compared to our baseline, which replicates the performance of previous state of the art model applied on OMI-DB, the proposed mass detection models achieve higher sensitivity and areas under the FROC curve.

Additionally, combining the predictions of RepPoints and Deformable DETR, with both Swin-T and Swin-B backbones, using weighted boxes fusion results in outperforming single model predictions and previous state of the art by a significant 5.7% on the area under the FROC curve. Furthermore, when combining these models predictions with those of their convolutional peers, the performance can be further improved by an additional 1.1%. The presented frameworks demonstrate the potential of transformer backbones in detection tasks on the medical imaging domain.

References

Agarwal, R., Díaz, O., Yap, M.H., Llado, X., Marti, R., 2020. Deep learning for mass detection in Full Field Digital Mammograms. Computers in biology and medicine 121, 103774.

Aly, G.H., Marey, M., El-Sayed, S.A., Tolba, M.F., 2021. YOLO Based Breast Masses Detection and Classification in Full-Field Digital Mammograms. Computer Methods and Programs in Biomedicine 200, 105823. URL: https://www.sciencedirect.com/science/article/pii/S0169260720316564, doi:https://doi.org/10.1016/j.cmpb.2020.105823.

Aristokli, N., Polycarpou, I., Themistocleous, S., Sophocleous, D., Mamais, I., 2022. Comparison of the diagnostic performance of Magnetic Resonance Imaging (MRI), ultrasound and mammography for detection of breast cancer based on tumor type, breast density and patient's history: A review. Radiography .

Balleyguier, C., Kinkel, K., Fermanian, J., Malan, S., Djen, G., Taourel, P., Helenon, O., 2005. Computer-aided detection (CAD) in mammography: does it help the junior or the senior radiologist? European journal of radiology 54, 90–96.

Berment, H., Becette, V., Mohallem, M., Ferreira, F., Chérel, P., 2014. Masses in mammography: What are the underlying anatomopathological lesions? Diagnostic and Interventional Imaging 95, 124–133. URL: https://www.sciencedirect.com/science/article/pii/S2211568413003872, doi:https://doi.org/10.1016/j.diii.2013.12.010. radio-histological correlations in breast imaging.

Broeders, M., Moss, S., Nyström, L., Njor, S., Jonsson, H., Paap, E., Massat, N., Duffy, S., Lynge, E., Paci, E., 2012. The impact of mammographic screening on breast cancer mortality in Europe: a review of observational studies. Journal of medical screening 19, 14–25.

Cao, H., Pu, S., Tan, W., Tong, J., 2021. Breast mass detection in digital mammography based on anchor-free architecture. Computer Methods and Programs in Biomedicine 205, 106033. URL: https://www.sciencedirect.com/science/article/pii/S0169260721001085, doi:https://doi.org/10.1016/j.cmpb.2021.106033.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers, in: European conference on computer vision, Springer. pp. 213–229.

CDC, 2022. "Breast Cancer Screening Guidelines for Women". https://www.cdc.gov/cancer/breast/pdf/breast-cancer-screening-guidelines-508.pdf. Accessed: 2022-05-20.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al., 2019. MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 .

DeSantis, C.E., Bray, F., Ferlay, J., Lortet-Tieulent, J., Anderson, B.O., Jemal, A., 2015. International variation in female breast cancer incidence and mortality rates. Cancer Epidemiology and Prevention Biomarkers 24, 1495–1506.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

D'Orsi, C., Bassett, L., Feig, S., et al., 2018. Breast imaging reporting and data system (BI-RADS). Breast imaging atlas, 4th edn. American College of Radiology, Reston .

Halling-Brown, M.D., Warren, L.M., Ward, D., Lewis, E., Mackenzie, A., Wallis, M.G., Wilkinson, L.S., Given-Wilson, R.M., McAvinchey, R., Young, K.C., 2020. OPTIMAM Mammography image database: a large-scale resource of mammography images and clinical data. Radiology: Artificial Intelligence 3, e200103.

Hassan, N.M., Hamad, S., Mahar, K., 2022. Mammogram breast cancer CAD systems for mass detection and classification: a review. Multimedia Tools and Applications , 1–33.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hodler, J., Kubik-Huch, R.A., von Schulthess, G.K., 2019. Diseases of the Chest, Breast, Heart and Vessels 2019-2022: Diagnostic and Interventional Imaging .

Ke, L., Mu, N., Kang, Y., 2010. Mass computer-aided diagnosis method in mammogram based on texture features, in: 2010 3rd International Conference on Biomedical Engineering and Informatics, IEEE. pp. 354–357.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context, in: European conference on computer vision, Springer.

pp. 740–755.

Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., et al., 2021a. Swin Transformer V2: Scaling Up Capacity and Resolution. arXiv preprint arXiv:2111.09883 .

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021b. Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.

Mann, R., Balleyguier, C., Baltzer, P., Bick, U., Colin, C., Cornford, E., Evans, A., Fallenberg, E., Forrai, G., Fuchsjäger, M., Gilbert, F., Helbich, T., Heywang-Köbrunner, S., Camps-Herrero, J., Kuhl, C., Martincich, L., Pediconi, F., Panizza, P., Pina, L., Pijnappel, R., 2015. Breast MRI: EUSOBI recommendations for women's information. European radiology 25, 3669–3677.

Mughal, B., Sharif, M., Muhammad, N., 2017. Bi-model processing for early detection of breast tumor in CAD system. The European Physical Journal Plus 132, 1–14.

Nickson, C., Mason, K.E., English, D.R., Kavanagh, A.M., 2012. Mammographic screening and breast cancer mortality: a case–control study and meta-analysis. Cancer Epidemiology and Prevention Biomarkers 21, 1479–1488.

Patel, B.C., Sinha, G., Soni, D., 2019. Detection of masses in mammographic breast cancer images using modified histogram based adaptive thresholding (MHAT) method. International Journal of Biomedical Engineering and Technology 29, 134–154.

Punitha, S., Amuthan, A., Joseph, K.S., 2018. Benign and malignant breast cancer segmentation using optimized region growing technique. Future Computing and Informatics Journal 3, 348–358.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28.

Ribli, D., Horváth, A., Unger, Z., Pollner, P., Csabai, I., 2018. Detecting and classifying lesions in mammograms with deep learning. Scientific reports 8, 1–7.

Sankatsing, V.D., van Ravesteyn, N.T., Heijnsdijk, E.A., Looman, C.W., van Luijt, P.A., Fracheboud, J., den Heeten, G.J., Broeders, M.J., de Koning, H.J., 2017. The effect of population-based mammography screening in Dutch municipalities on breast cancer mortality: 20 years of follow-up. International Journal of Cancer 141, 671–677.

Shamshad, F., Khan, S., Zamir, S.W., Khan, M.H., Hayat, M., Khan, F.S., Fu, H., 2022. Transformers in Medical Imaging: A Survey. arXiv preprint arXiv:2201.09873 .

Solovyev, R., Wang, W., Gabruseva, T., 2021. Weighted boxes fusion: Ensembling boxes from different object detection models. Image and Vision Computing 107, 104117.

Sun, P., Zhang, R., Jiang, Y., Kong, T., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C., et al., 2021. Sparse r-cnn: End-to-end object detection with learnable proposals, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14454–14463.

Sung, H., Ferlay, J., Siegel, R.L., Laversanne, M., Soerjomataram, I., Jemal, A., Bray, F., 2021. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: a cancer journal for clinicians 71, 209–249.

Wang, Z., Yu, G., Kang, Y., Zhao, Y., Qu, Q., 2014. Breast tumor detection in digital mammography based on extreme learning machine. Neurocomputing 128, 175–184.

World Health Organization, 2022. Breast cancer. https://www.who.int/news-room/fact-sheets/detail/breast-cancer. Accessed: 2022-05-20.

Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S., 2019. Reppoints: Point set representation for object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9657–9666.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum,

H.Y., 2022. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 .

Zhu, C., He, Y., Savvides, M., 2019. Feature selective anchor-free module for single-shot object detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 840–849.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J., 2020. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 .

Zoph, B., Cubuk, E.D., Ghiasi, G., Lin, T.Y., Shlens, J., Le, Q.V., 2020. Learning data augmentation strategies for object detection, in: European conference on computer vision, Springer. pp. 566–583.

Detection of Cellular Events in DIC Live Microscopy Videos

Aroj Hada[a], Mario Guarracino[c], Lucia Maddalena[c]

[a]MedicAl Imaging And Applications (MAIA)
[b]University of Cassino
[c]National Research Council, Italy (CNR)

Abstract

Time-lapse microscopy of cells is a routinely performed experiment in many biology laboratories, which serve key importance in many applications, particularly when studying dynamic processes such as drug response and cancer studies. Some key events occur during the cell cycle under normal or perturbed conditions, including i) Mitosis, ii) Apoptosis, iii) Multipolar Division, and iv) Failure of Cell Division. Automatic identification and localization of these events within the videos is of key importance to obtain useful information on many laboratory experiments. Here, we present a differential interference contrast (DIC) live-cell microscopy dataset consisting of such events with the goal of localization and classification of these events within the video frames. We propose an object detection-based approach for the task by treating each event instance as an object. Several object detection algorithms have been applied to assess the performance for the detection and classification of three key events; i) Early Mitosis, ii) Late Mitosis, and iii) Apoptosis. YOLOv5 model achieves the best detection results, reaching mAP@0.5 (Mean Average Precision) scores of 0.943 on the test set.

Keywords: DIC, Object Detection, Time-lapse microscopy, Mitosis, Apoptosis

1. Introduction

Advances in Computer vision, Machine Learning, and Deep Learning algorithms have made significant progress in the active field of event detection in videos. Many techniques have been developed to tackle the various problems associated with the field, such as abnormality detection for surveillance applications and human action detection.[Huh (2013)] However, one important application has not been given its deserved attention - cellular event detection in time-lapse images from transmitted light cell microscopy.

Time-lapse microscopy imaging is being used in an increasing number of biological and biomedical studies to observe the dynamic behavior of cells over time which help quantify important data, such as the number of cells and their sizes, shapes, and dynamic interactions across time. These quantitative properties provide critical insight into the fundamental nature of cellular function [Jiang et al. (2020)]. Because of this, live-cell imaging has become a requisite analytical tool in most cell biology laboratories, as well as a routine methodol-

ogy that is practiced in the wide-ranging fields of neurobiology, developmental biology, pharmacology, and many other related biomedical research disciplines.

One of the major purposes for monitoring a cell population is to study single-cell behavior in response to physiological or external stimuli and understand the underlying mechanisms. For example, in drug discovery and cancer research, Naso et al. (2020) have used time-lapse microscopy to look at cell response to anti-mitotic drugs in terms of cell division and cell death. To achieve this goal, quantitative information on cell behavior needs to be obtained and analyzed. Among various cell behaviors, the behavior regarding proliferation and fate are of main importance. All this is usually done manually with protocols such as Caldon and Burgess (2019). Therefore, automated systems for detecting cellular events such as mitosis (cell division) and apoptosis (cell death) are of great interest.

This proposal investigates approaches to automatically localizing and classifying these cellular events with deep learning algorithms. We more specifically describe our problems for given time-lapse live-cell mi-

croscopy images as follows:

- Mitosis detection identifies when (at which frame) and where (at which x and y positions) cells enter and end the mitotic state in the time-lapse images.

- Apoptosis detection identifies when and where a cell death event occurs in the time-lapse images.

- Daughter cell detection identifies when and where mitotic cells in early mitosis phase divide into daughter cells (telophase and anaphase) in the time-lapse images.

- Multipolar division detection identifies when and where a multipolar division occurs in the time-lapse images. A multi-polar division occurs when a single mitotic cell divides into more than two cells. The most common example of this is the tripolar case where three daughter cells emerge from a single mother cell.

- Failure of division detection identifies when and where abnormal mitosis cases where mitotic cells fail to divide into daughter cells and go back into the media as a single interphase cell in the time-lapse images.

A visualization of all of the above-mentioned events are presented in Fig. 11 in the form of single-cell sequential image patches and in Fig. 3 in the form of individually identified phenotypes.

Time-lapse live-cell microscopy, such as phase-contrast microscopy and differential interference contrast (DIC) microscopy, enables long-term monitoring of live and intact cells. Most current high-throughput microscopy imaging approaches resort to the use of cellular staining, with which only short-term monitoring of cells is allowed due to the photo-toxicity of reagents used. One way to minimize this photo-toxicity is to work with bright-field or transmitted light techniques rather than fluorescence [Brown (2014)].

1.1. Transmitted Light Microscopy

Transmitted light microscopy is the general term used for any type of microscopy where the light is transmitted from a source on the opposite side of the specimen from the objective. The microscopic techniques requiring a transmitted light path include brightfield, darkfield, Zernicke phase (or just phase), and DIC (or Nomarski) [Lang (1982)] optics. Examples are provided in Fig. 1.

DIC is an imaging technique for rendering contrast in transparent specimens used for imaging live and unstained biological samples, such as a smear from a tissue culture. Its resolution and clarity for imaging such biological samples are unrivaled among standard optical
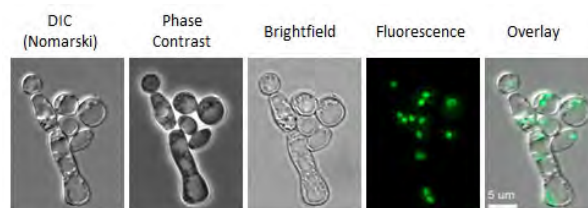
Figure 1: Different microscopy imaging modalities.

microscopy techniques. Image quality, when used under suitable conditions, is outstanding in resolution and almost entirely free of artifacts, unlike phase-contrast.

Phase-contrast and DIC microscopy are complementary techniques capable of producing high-contrast images of transparent biological phases that do not ordinarily affect the amplitude of visible light waves passing through the specimen [Rosenthal (2009)]. Phase-contrast produces images with bright objects on a medium gray background, while DIC produces relatively transparent gray objects in a gray background. DIC imaging possesses several advantages compared to phase-contrast in biological, usage of equipment, absence of artifacts such as the halo effect, and can produce excellent high-resolution images [Murphy et al. (2017)].

2. State of the art

Event detection in transmitted light time-lapse microscopy images is an application still in its infancy. Although some works have been done to address the issue, there is no so-called state-of-the-art, mainly due to a lack of sufficient curated and annotated public datasets and a lack of consensus on the evaluation of these methods. Furthermore, the task has been broken down into the detection of specific events individually, such as mitosis detection or cell death detection, in order to counter the absence of datasets. Meanwhile, few to no papers have worked on cases such as multipolar division or failure of division, even though these events are routinely detected in laboratory settings and are manually recorded.

Su et al. (2017) and Mao and Yin (2017) proposed a convolutional long short-term memory (CNN-LSTM) network and a Two stream Bidirectional CNN-LSTM network on sequences of single-cell image patches and utilized both spatial and temporal information in order to detect mitosis events. They report an average precision of 0.96 and 0.98, respectively. However, these models utilize a large amount of manually annotated data to train on, and both papers also report a sharp decrease in accuracy when testing the model on other cell datasets.

Lu et al. (2018) put forward a Time-lapse Microscopy image in Nanowell Grids (TIMING) dataset for label-free apoptosis classification using CNN and LSTM

Models. The dataset contained single-cell crops, and were classified as live or dead cells.

A CNN-LSTM model that learns spatial and temporal locations of the cells from a detection map in a semi-supervised manner was proposed by Phan et al. (2019) for the detection of mitosis in phase-contrast videos. The method mentions the use of only 1050 annotated frames to achieve an F1 score of 0.544-0.822 depending on the video. However, it also mentions the decrease in performance with the increase in the input sequence length, which is not ideal for practical situations where time-lapse experiment's video sequences may contain thousands of frames. The method also will only be able to detect a single event at a time, such as mitosis, whereas these events can randomly occur in multiple places in a single frame.

An object detection method was proposed by Von Chamier et al. (2020) in the paper "Democratising deep learning for microscopy with ZeroCostDL4Mic". Here the authors detected and localized the events present in time-lapse microscopy videos using YOLOv2 architecture and were able to achieve a mAP@0.5 of 0.60 for their own presented dataset. The paper reflects on easing the use of Deep Learning approaches for microscopic data analysis.

Nishimura and Bise (2020) proposed a method for multiple mitosis event detection and localization by estimating a spatial-temporal likelihood map using a 3DCNN architecture (V-Net). In the likelihood map, a mitosis position is represented as an intensity peak with a Gaussian distribution, in which multiple mitoses are represented as multiple peaks. The proposed method had an average precision of 0.862 on their own dataset.

While the method does take into account the spatial and temporal information, it is only limited to the detection of mitosis events and not any other events that may be associated with mitosis. In order to identify other events as well, multiple models based on this method would be needed. Furthermore, usage of this method for other datasets or cell lines requires the generation of laborious manual annotations in the form of Gaussian distributed likelihood maps.

La Greca et al. (2021) demonstrated the use of classical DL approaches like ResNet over transmitted light microscopy (TLM) cell death, where they have classified different cell lines as dead or alive by using complete frames as input images. On images that contained both alive and dead cells, the model was able to predict the dead cells, which were localized by looking at the class activation maps (CAM) that reconstruct heat map-like visualizations merging the information provided by the last convolutional layer and the model predictions. These predictions were compared with human performance and were found to largely outperform human ability.

Table 1 summarizes all deep learning based work in Cellular event detection. Even though the above-mentioned methods have several positives, they also have several issues, among which the major ones can be listed as:

- Most of the methods base their work on datasets containing images based on fluorescence microscopy or phase-contrast techniques where the cell background contrast is higher, and thus the cells are easily distinguishable from the background. The same is not true for DIC images,

Table 1: Comparison of previous works in cellular event detection

Author	Method	Dataset	Image Modality	Dataset Size	Detected Events	Metric	Metric-values
Su et al. (2017)	CNN-LSTM	Private	PC	2000 event videos	Mitosis	F1-score	0.97
Mao and Yin (2017)	TS-BLSTM	Private	PC	500 event videos	Mitosis	Precision-Recall	0.98-0.97
Lu et al. (2018)	CNN-LSTM	Deep-TIMING (Not-available)	PC	72000 cropped cells	Cell Death	Precision-Recall	-
Phan et al. (2019)	Unsupervised CNN-LSTM	Not available	PC	1050 frames	Mitosis	F1-score	0.544-0.822
Von Chamier et al. (2020)	YOLOv2	Public	DIC	40 frames	Mitosis	mAP	0.60
Nishimura and Bise (2020)	V-Net	CVPR2019 mitosis detection (Not-available)	PC	1013 images	Mitosis	Precision	0.862
Cell-Death, La Greca et al. (2021)	ResNet	Public	PC	14k images	Cell Death	Accuracy	0.64 (for u2os cell-lines)

where the cells are also transparent and seem to be essentially indistinguishable from the background upon high confluence.

- The methods usually identify themselves as event detection but in truth, they are tailored to detection of only one of the events that may occur, i.e., Mitosis or Apoptosis. This may be attributed to the lack of curated datasets available.

- The datasets used have not been made publicly available.

- Finally, the models usually are trained for a specific cell-line which when used for other cell-lines fail to generalize. Hence, to bring the work into application, there is a need to re-train the model with the new specific dataset. For models that require a large set of annotated data, this is no feasible.

3. Material and methods

3.1. Dataset

24 raw time lapse microscopy video frames of Human Bone Osteosarcoma Epithelial Cells (U2OS Line) based on Differential interference contrast (DIC) microscopy technique were generated at the Nikon imaging platform at IBPM-CNR of Rome. The videos consisted of multiple events in each frame including Cell division, Cell Death, Tripolar/Multipolar Division, and Failure of Cell Division.

Each video has 60-70 frames and each frame can consist from none to seven event instances of interesting events. Table 2 shows the exact number of videos and frames present for each event class. The event "Mitosis" can be extracted from all the videos as it is a prerequisite that the cells undergo early mitotic state and later diverge into different fates. Each video within it's frames may contain one or more cellular event other than mitosis.

Table 2: Number of videos per event

Event	No. of videos	No. of Frames
Apoptosis	9	598
Tripolar Division	10	542
Failure of Division	5	301
Mitosis	All	All
Total	24	1441

3.1.1. Image Acquisition

U2OS cell lines seeded in 2-4 micro-slides (Ibitreat) were observed with an inverted microscope (Eclipse Ti, Nikon) using a 40× (Plan Fluor, N.A. 0.60, DIC) or a 60× Oil (Plan Apo, N.A. 1.4, DIC) objective (Nikon).

During the whole registration, cells were kept in a microscope incubator (Basic WJ, Okolab) at 37°C in 5% CO_2. DIC images were acquired every 5 or 7 min using a DS-Qi1Mc camera (Nikon) or a Clara camera (ANDOR technology). Asynchronous cultures were treated with Aurora kinase inhibitor (MLN8237) to induce mitotic defects and cell death.

3.1.2. Image pre-processing

DIC imaging produces positive and negative peaks at the edges of cell structures, while unchanging structure results in a gray background intensity similar to that found outside the cell, where the internal structure of a cell has the same intensity as the image background [Furcinitti (2013)].

The gray images were subjected to multiple image processing techniques to see if the event-associated objects could be improved visually. However, because of the nature of the microscopy technique, which provides bright transparent objects on a bright background, most techniques failed. Only gamma correction and sharpening with a high-pass filter from OpenCV were applied to the images to obtain sharper objects with uniform illumination. Fig. 2 illustrates some example frames from the dataset before (a) and after (b) pre-processing.
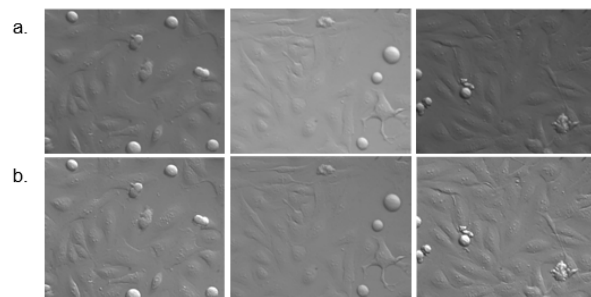
Figure 2: Examples of the DIC microscopy frames present in the dataset: a) original images and b) pre-processed images with gamma correction to get uniform illumination and image sharpening with a high-pass filter.

Contrast enhancement techniques such as CLAHE [Yadav et al. (2014)] did produce strong objects but it also resulted in highlighting the interphase cells present in the background that are not much of interest forward. Hence, contrast enhancement was not utilized. Furthermore, images provided were of size 400X320p. Necessary padding was done on the images to have a common height and width to 412X412p.

3.1.3. Data Annotation

For the object detection approach, each event instances present in the video frames were treated as individual objects and were annotated with bounding boxes through an open-source image annotation tool, "makesense.ai" Skalski (2019). The identification of these events was done based on the morphology of the cell
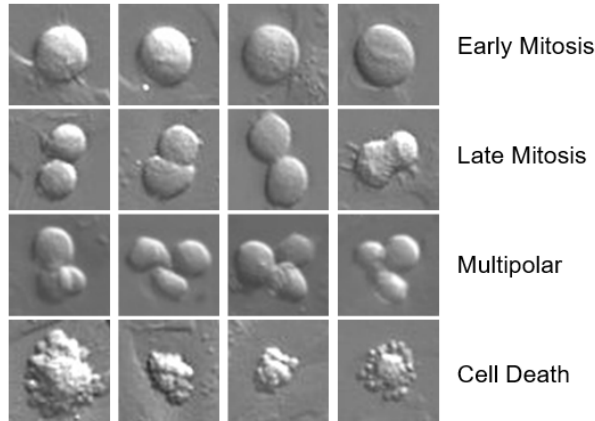
Figure 3: Examples of identified cell phenotypes and representative examples that were hand-labelled to build the training dataset from bright-field time-lapse videos.

Figure 4: One of the frames annotated with bounding boxes to build the dataset from time-lapse videos. Each frame can consist of multiple events that occur independently. In this example, we find four independent events: one Early Mitotic (red box), one Cell Death/Apoptosis (cyan box), and two Late Mitotic Cells (blue boxes).

states with the help of an expert biologist. The primary annotated dataset consists of all the event classes present except for the "Failure of Division" class, since the class cannot be identified through phenotype alone and requires temporal information as well. An approach to identify the said event has been discussed in the limitations section 4.2 of the Results and Discussion.

The event instances were annotated as four different classes; Mitosis as two separate classes i) Early Mitosis (Mitotic Circular) and ii) Late Mitosis (Dividing), iii) Apoptosis (Cell Death), and iv) Multipolar (Tripolar Division). A visualization of these independent event instances is presented in Fig. 3, while Fig. 4 displays a frame annotated with bounding boxes for different classes. In order to keep the naming conventions simple, alternative less technical class names have been used synonymously throughout this document. The class names included in the parenthesis represent these names.

Annotations were obtained in Pascal VOC format with xml files for each images and were later converted into other formats such as COCO JSON, YOLO format, etc as required. The conversions were done using Moore and Corso (2020) library.

3.1.4. Dataset Preparation

The distribution of the annotated objects per class can be seen in Fig. 5A. There is a large class imbalance within the dataset between the largest, Early Mitosis, and the smallest class, Tripolar Division. Therefore, we tried to improve the class distribution by class balancing techniques. Namely, we applied undersampling of the majority class by removing some of the images including objects belonging to the Early Mitosis class. Moreover, we applied oversampling of the minority class, applying augmentation techniques (i.e., horizontal and vertical flipping, and rotations) to the images including objects belonging to the Tripolar Division class. The
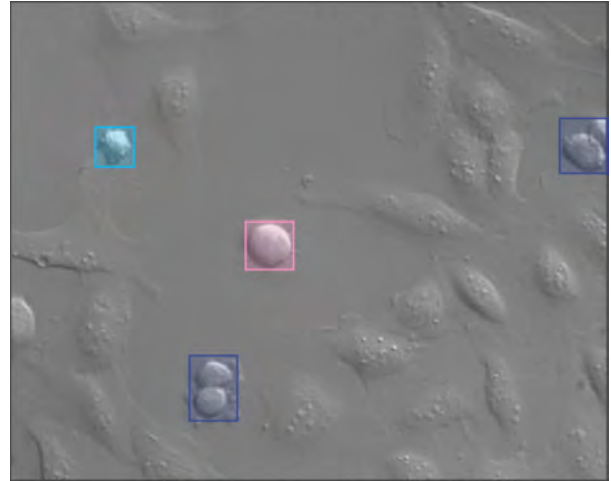
resulting class distribution of the complete dataset is shown in Fig. 5A.

Initial tests were performed on the complete 4-class dataset using YOLOv5s as a benchmark model (see Table 3 for details). The extreme poor performance, not only concerning the minority Tripolar Division class, but all the classes, lead us to drop the "Tripolar Division" class. Excluding from the dataset all the images containing this type of event, we obtained a 3-class dataset consisting of 683 images with a total of 714 Early Mitosis, 290 Late Mitosis, and 531 Cell Death events.

Furthermore, a number of background images, i.e., images with no objects, were also included in the dataset amounting to approximately 10% of the complete final dataset in order to reduce False Positive cases

The 3-class dataset was split into different train, validation and test sets such that the frames from the same video would not be present in different splits The distribution of this final 3-class dataset can be seen in Fig. 5B.

3.2. Applied Methods

Based on the literature review of previous work done for event detection in time-lapse microscopy videos, we propose using each individual phenotype event present in the frames as an object, essentially reducing the problem to a classical object detection and classification challenge. This approach was chosen with various factors in mind, especially the small size of the dataset, lack of annotations on the data, the requirement of refining the dataset, and most importantly, ease of use of the application by the biologist.

Object detectors today usually can be divided into two parts, a backbone that is pre-trained on ImageNet
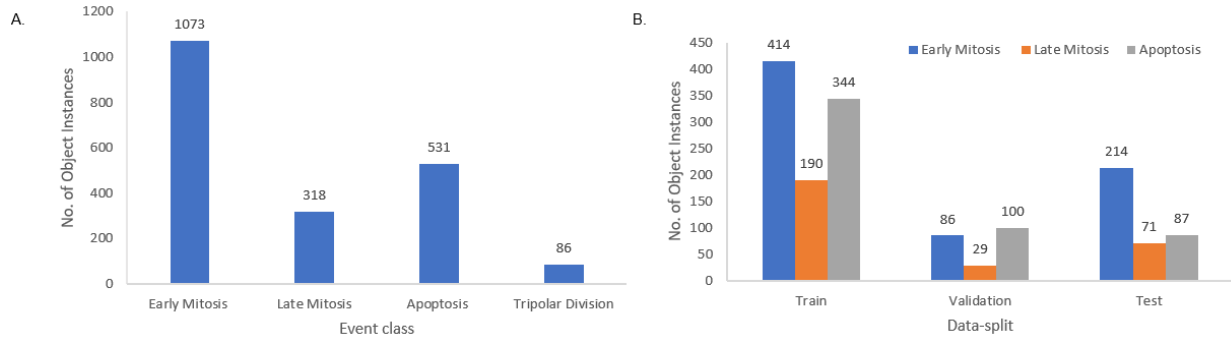
Figure 5: Distribution of object instances per class in: A) the complete 4-class dataset and B) the 3-class dataset with train-val-test splits.

and a head which is used to predict object classes and object bounding boxes. The head can be further classified into two types, one-stage object detectors such as YOLO, SSD, and RetinaNet, and two-stage object detector such as the R-CNN family. Object detectors in recent years often include some layers that connect the backbone and the head, called the neck, which are usually used to collect feature maps from different stages. Fig. 6 illustrates the common structure for these object detection frameworks. Additionally, there also have been some other architectures that put an emphasis on directly building a new backbone or a completely new model structure.

Several different object detection models were used to achieve the solution with multiple experiments within each model.

3.2.1. Faster-RCNN

First proposed by Ren et al. (2015), Faster-RCNN is the most widely used state-of-the-art version of the R-CNN family. These networks usually consist of — a) A region proposal algorithm to generate "bounding boxes" or locations of possible objects in the image; b) A feature generation stage to obtain features of these objects, usually using a CNN; c) A classification layer to predict which class this object belongs to; and d) A regression layer to make the coordinates of the object bounding box more precise. The Faster R-CNN algorithm improves upon the selective search algorithm by introducing another convolutional network, Region Proposal Network (RPN) to generate the region proposals. Hence, Faster R-CNN can be summarised as a detection pipeline that uses the RPN as a region proposal algorithm, and Fast R-CNN as a detector network. A Non-maxima suppression (NMS) Bodla et al. is also applied with a threshold. From the top down, all of the bounding boxes which have an IoU of greater than the threshold with another bounding box are discarded. Thus the highest-scoring bounding box is retained for a group of overlapping boxes.

3.2.2. RetinaNet

RetinaNet is a one-stage object detection model that utilizes a focal loss function to address class imbalance during training. Proposed by Lin et al. (2017) of Facebook AI Research (FAIR) in the paper titled "Focal Loss for Dense Object Detection", they showcased the application of a new loss function called the Focal loss. Focal loss applies a modulating term to the cross entropy loss in order to focus learning on hard negative examples. RetinaNet is a single, unified network composed of a backbone network and two task-specific subnetworks. The backbone is responsible for computing a convolutional feature map over an entire input image and is an off-the-self convolutional network. The first subnet performs convolutional object classification on the backbone's output; the second subnet performs convolutional bounding box regression. The two subnetworks feature a simple design that the authors propose specifically for one-stage, dense detection.

3.2.3. YOLO Architechtures

Models based on YOLO (You Only Look Once) use a single neural network to process an entire picture, then separate it into a grid system and predict bounding boxes and probabilities within each grid. These methods are "just looks once" at the image in the sense that they make predictions after only one forward propagation run through the neural network. They then deliver detected items after non-maxima suppression.

YOLOv2 Redmon and Farhadi (2017) is a single-stage real-time object detection model. It improves upon YOLOv1 in several ways, including the use of Darknet-19 as a backbone, batch normalization, use of a high-resolution classifier, and the use of anchor boxes to predict bounding boxes, and more. YOLOv2 achieved the state-of-the-art (SOTA) title for general object detection and classification task in 2016 outperforming the previous SOTA models like Faster-RCNN and RetinaNet on the combined COCO 2007 and 2012 dataset while still running significantly faster.

YOLOv4 is one of the newer additions to the YOLO series, published in 2020 by Bochkovskiy et al. (2020).
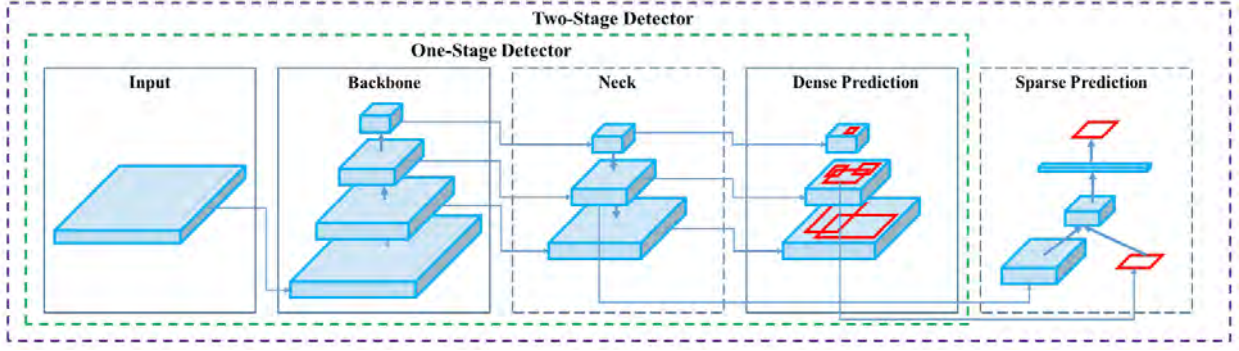
Figure 6: Object detector frameworks. (Extracted from Bochkovskiy et al. (2020))

This framework introduced what they called "Bag of freebies" and "Bag of Specials" additional methods of data augmentation Mosaic and Self-Adversarial Training (SAT). Mosaic creates a mix of four different training images into one. Self-Adversarial Training operates in two forward and backward stages. In the first stage, the network alters the only image instead of the weights. In the second stage, the network is trained to detect an object on the modified image.

While YOLOv4 is based on the Darknet framework written in C, a version of the same algorithm based on the PyTorch framework was released by Jocher et al. (2022). This framework is called YOLOv5 and resembles YOLOv4 completely in its implementation otherwise. YOLOv5 also includes additional methods, called hyperparameter evolution, that help to perform hyperparameter optimization on the model. With these optimized hyperparameters, training a new model results in increased performance. YOLOv5 within itself contains multiple models, named according to their depth and number of parameters: YOLOv5s (Small), YOLOv5m (Medium), YOLOv5l (Large), and YOLOv5xl(XLarge).

3.2.4. YOLOX

YOLOX is a single-stage anchor-free object detector proposed by Ge et al. (2021). It makes improvements on previous iterations of the YOLO series with introduction of three fundamental innovations. First, it employs a decoupled head where classification and localization operations are separated instead of a coupled head. Next is the use of anchor-free boxes by reducing the number of predictions for each location from 3 to 1 and selection of 1 positive sample for each object. Lastly, it utilizes SimOTA for label assignment where label assignment is formulated as an optimal transport problem via a top-k strategy. Additionally, YOLOX uses augmentation strategies such as adding Mosaic and MixUp to boost performance.

3.2.5. VFNet

Introduced by Zhang et al. (2021), VarifocalNet is a method aimed at accurately ranking a huge number of candidate detections. It consists of a new loss function, named Varifocal Loss, for training a dense object detector to predict the IoU-aware Classficiation Score (IACS), and a new efficient star-shaped bounding box feature representation for estimating the IACS and refining coarse bounding boxes. Combining these two new components and a bounding box refinement branch, results in a dense object detector, what the authors call VarifocalNet or VFNet for short. Varifocal loss is based on the binary cross entropy loss as well as Focal loss and is defined as:

$$VFL(p, q) = \begin{cases} -q(qlog(p) + (1-q)log(1-p)) & q > 0 \\ -\alpha p^{\gamma} log(1-p) & q = 0 \end{cases}$$

where p is the predicted IACS and q is the target score. For a foreground point, q for its ground-truth class is set as the IoU between the generated bounding box and its ground truth (gt IoU) and 0 otherwise, whereas for a background point, the target q for all classes is 0.

3.3. Implementation Details

The above-mentioned models were implemented using different Python libraries as each model may have its own implementation workflow. Multiple experiments were carried out for each model to obtain the highest metric values possible, but only experiments that reflect some insights are mentioned. All models were trained using pre-trained weights provided by the respective framework used. Multiple experiments were run for each model at intervals of 30, 100, 200, and 300 epochs to determine the least number of epochs required to achieve the best metrics. A maximum Batch size was used for each model, which can be found in Table 4. The imgaug library [Jung et al. (2020)] was used to augment the training set images and the bounding boxes by random rotation, flipping, and brightness/contrast augmentations for the YOLOv2 and the VFnet models, while

for the rest of the models, image augmentation was performed using the built-in config files. The transform probabilities were made sure to be the same in all the runs.

- The YOLOv2 model was trained based on an implementation provided by Von Chamier et al. (2020).

- The Detectron 2 library from Facebook AI Research was used to train RetinaNet and Faster-RCNN implementations. Faster-RCNN models were trained with a ResNeXt-101 backbone pretrained on COCO dataset, and RetinaNet models were trained with a ResNet-101 backbone pretrained on COCO dataset.

- Both YOLOv5 and YOLOX models were implemented using their official GitHub repositories. For YOLOv5, from Ultralytics as suggested by Jocher et al. (2022) and for YOLOX, from Megvii-BaseDetection as proposed in Ge et al. (2021). Pretrained weights from the COCO dataset were used to initialize both models. On the best YOLOv5 model obtained, hyperparameter optimization was run for 300 generations using the hyperparameter evolution method presented in YOLOv5.

- VFnet was implemented through Chen et al. (2019) MMDetection library from OpenMMLab. The model was trained with a ResNet-101 backbone with weights initialized on the COCO dataset.

All experiments were performed either on a machine with 32 GB RAM and NVidia GeForce RTX 2060 GPU with 8 GB memory or on NVIDIA P100 with 25 GB of RAM.

3.4. Evaluation Metrics

Object detection models are usually evaluated using metrics such as Average Precision (AP) and mean Average Precision (mAP). Along with precision, the recall metric is also considered important as it gives insights into false positives that the model may predict. The MI-DOG2021 mitosis detection challenge in histopathology images in MICAAI 2021 presented by Aubreville et al. (2022) used F1 score as the main main metric to evaluate model performance which takes into account for false positives as well as false negatives and is thus considered to be a fair metric.

3.4.1. Intersection over Union (IoU)

The IoU metric in object detection evaluates the degree of overlap between the ground (*gt*) truth and prediction (*pd*) bounding boxes. IoU ranges between 0 and 1, where 0 shows no overlap and 1 means perfect overlap between *gt* and *pd*. It is calculated as follows:

$$IoU = \frac{area(gt \cap pd)}{area(gt \cup pd)}$$

IoU tresholding can then be used to decide if a detection is correct or not. For a given IoU threshold α, a True Positive (TP), i.e., a correct positive prediction, is a detection for which $IoU(gt, pd) \geq \alpha$ and a False Positive (FP), i.e., a wrong positive detection, is a detection for which $IoU(gt, pd) < \alpha$. A False Negative (FN) is an actual instance that is not detected.

3.4.2. Precision, Recall, and F1

Precision is the degree of exactness of the model in identifying only relevant objects. It is the ratio of TPs over all detections made by the model:

$$Precision = \frac{TP}{TP + FP}$$

Meanwhile, Recall measures proposition of TPs among all ground truths. It gives the percentage of detected true positives as compared to the total number of true positives in the ground truth. Mathematically, It is the ratio of TPs over all ground truth objects and is defined as:

$$Recall = \frac{TP}{TP + FN}$$

Using these metrics, generally, a method is considered good if it reaches high Recall values, without sacrificing Precision. The F-measure (F1) is a metric that combines Precision and Recall, given by their weighted harmonic mean:

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision}$$

3.4.3. Average Precision & Mean Average Precision (mAP)

AP@α is Area Under the Precision-Recall Curve (AUC-PR) evaluated at α IoU threshold. Formally, it is defined as follows

$$AP@\alpha = \int_0^1 p^r \, dr$$

AP is calculated individually for each class. This means that there are as many AP values as the number of classes (loosely). These AP values are averaged to obtain the mean Average Precision metric. Precisely, mean Average Precision (mAP) is the average of AP values over all classes.

$$mAP@\alpha = \frac{1}{n} \sum_{i=1}^{n} AP_i \quad \text{for } n \text{ classes}$$

All of these metrics were calculated on the validation and test set using evaluation scripts present within each algorithm as well as a separate evaluation script from Padilla et al. (2021). All the evaluations were performed on predictions made with a confidence score of 0.25 from each of the models.

6.8

3.5. Application

One of the main goals of the project was to provide an ease of implementation of the proposed method to the biologist. After a lot of considerations on the pros and cons for methods for this purpose, such as GUI desktop application and web-based Flask implementation [Grinberg (2018)], a fully customizable google co-lab notebook was prepared with the best performing model, keeping in mind the ease of use. This was a similar approach as the one taken by Von Chamier et al. (2020), where the biologist would be able to easily to train the state-of-the-art models on their own dataset.

4. Results

As mentioned earlier, all experiment setups for each model utilized either the original 4-class dataset or the 3-class dataset, both of which include evaluation on a validation as well as a test dataset. The presented results are structured in three subsections: first the results on the 4-class dataset are shown, then the results on the 3-class dataset are presented, finishing with detailed results using the best deep learning approach.

4.1. 4-class dataset

On the 4-class dataset with augmentation for the minority class, the highest mAP@0.5 of 0.680 was achieved using the YOLOv5m benchmark model. As suggested by all the performance values, reported in Table 3, as well as by the qualitative visualization of the classification results, the model was not able to perform well on the 4-class dataset, even when we augment the minority class, Tripolar Division.

4.2. 3-class dataset

On the final 3-class dataset prepared as mentioned in section 3, all the methods seem to perform fairly well. Among the several methods applied, the worst performance was achieved by the Faster-RCNN model, while the highest mAP@0.5 was achieved by the YOLOv5m with optimized hyperparameters. For this model, the use of hyperparameter evolution helped increase the mAP@0.5 score from 0.923 in the default YOLOv5m model to 0.943. The best performance values obtained from each model are reported in Table 4. Values for some of the metrics in the table could not be reported as some models did not explicitly provide precision and recall values. The per-class mAP scores are provided in Table 5. Among all the models, the highest AP achieved was for the Early Mitosis event class, followed by the Apoptosis class.

Relatively newer architectures, such as YOLOX and VFNet, were expected to achieve higher performance compared to the other models. However, this was not the case. Both the models, proposed in 2021,

reached a lower F1-score compared to their predecessor, YOLOv5. Although YOLOX models performed exceptionally well on the validation set compared to YOLOv5, on the test set the metric values for YOLOX models decreased.

A quick comparison with the previous works in the field mentioned in Table 1 reveals that the proposed algorithms, even without any temporal information inclusion, can perform as well as some of the SOTA models that utilize spatio-temporal information.
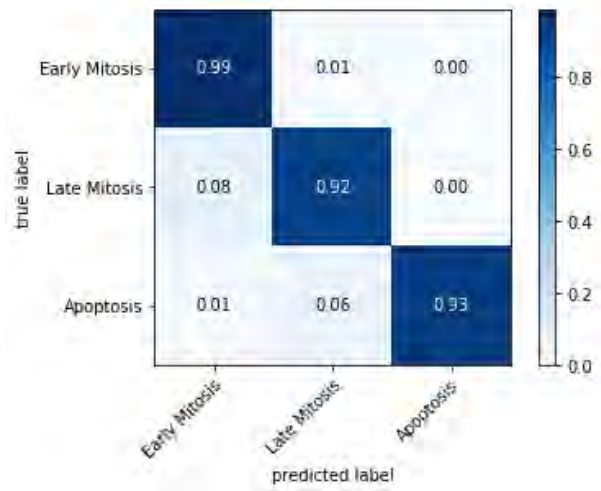
Figure 7: Confusion matrix for the YOLOv5m model on the test set of the 3-class dataset.
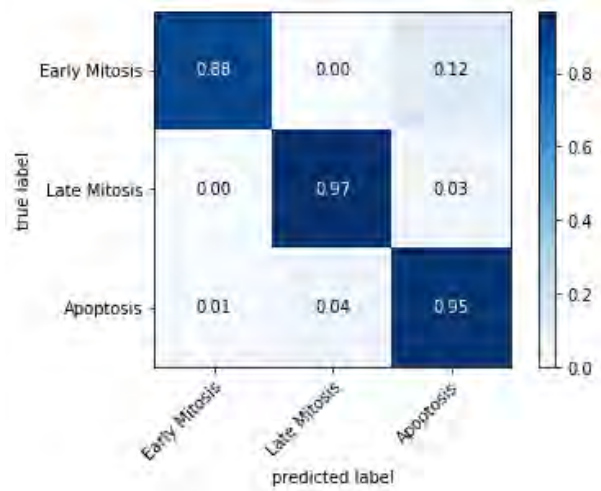
Figure 8: Confusion matrix for the YOLOv5m model on the validation set of the 3-class dataset.

Figures 7 and 8 present the confusion matrices obtained from the best performing model, YOLOv5m, for the validation and the test dataset, respectively. The heatmaps represent intense colors when the predicted and actual results ratio is close to 1. It can be clearly seen that all the three classes have intense colors representing high likeness among the predicted and actual

Table 3: Performances on the test set for the 4-class dataset

Approach	mAP@0.5	Precision	Recall	F1-score	Batch size	No. of Epoch
YOLOv5m	0.680	0.451	0.878	0.5959	16	30

Table 4: Comparison between different models on the test set for the 3-class dataset

Approach	mAP@0.5	mAP@0.75	mAP@0.5:0.95	Precision	Recall	F1	Batch size	No. of Epoch
Faster-RCNN	0.877	0.5656	0.523	0.8361	0.658	0.736	64	60
RetinaNet	0.897	0.5574	0.5393	0.91	0.743	0.818	64	60
YOLOv2	0.812	-	0.535	0.741	0.88	0.804	32	60
YOLOv5s	0.916	-	0.54	0.949	0.871	0.908	16	30
YOLOv5m	0.928	-	0.553	0.972	0.854	0.909	16	30
YOLOv5m hyp	**0.943**	-	**0.589**	0.945	0.911	**0.927**	16	30
YOLOv5x	0.922	-	0.546	0.916	0.837	0.874	16	30
YOLOX	0.838	0.57	0.53	0.822	0.6028	0.756	16	30
VFNet	0.899	0.5724	0.5315	0.873	0.677	0.762	16	30
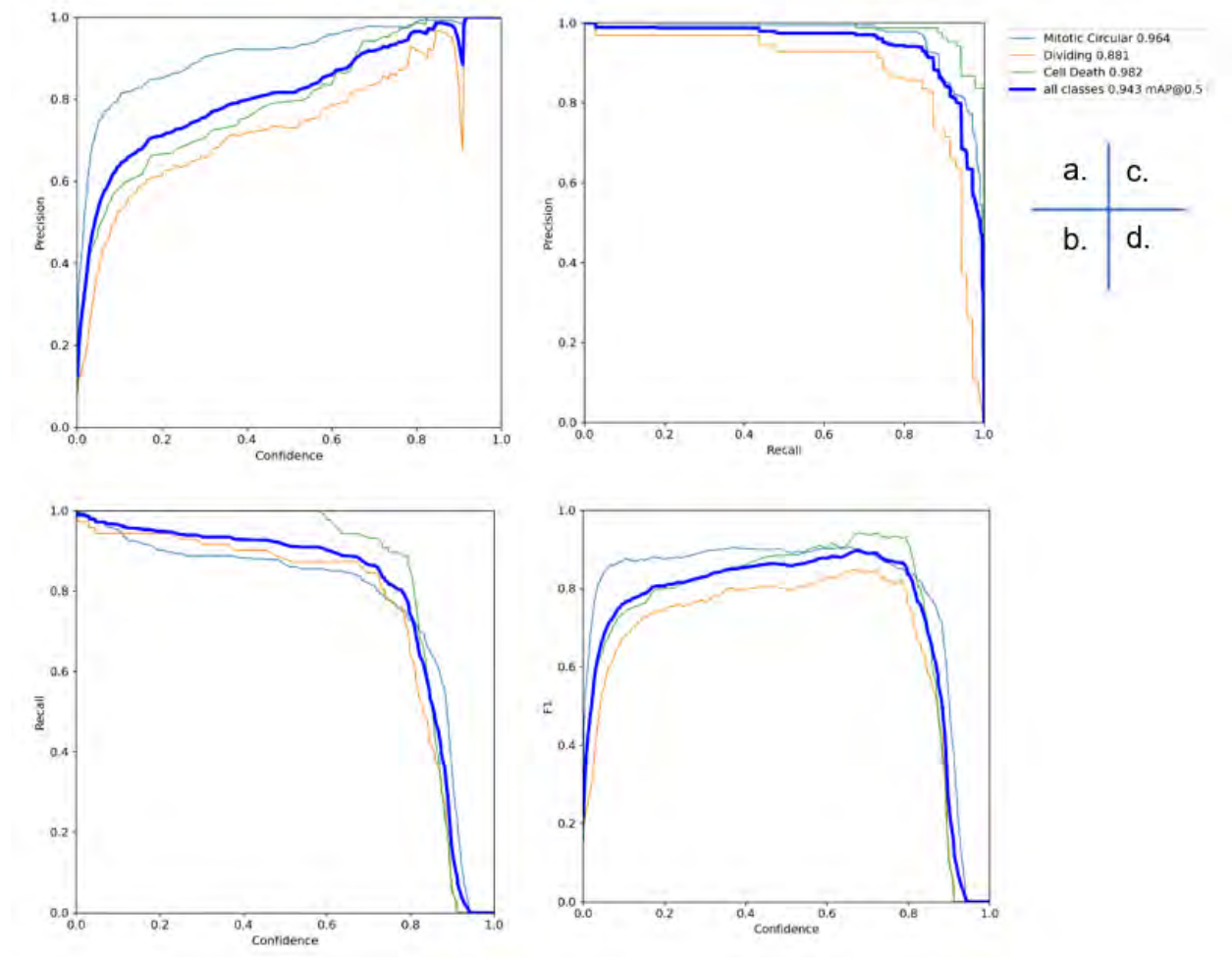
Figure 9: Plots showing Precision, Recall, and confidence values for each class obtained from the YOLOv5m predictions on the test dataset: a) Precision vs. confidence, b) Recall vs. confidence, c) Precision vs. Recall, and d) F1 vs. confidence. The curves represent all classes (Blue), Early Mitosis/Mitotic Circular (Cyan), Late Mitosis/Dividing (Orange), and Apoptosis/Cell Death (Green).

values for both the validation and the test datasets.

Figures 14 and 15 in Appendix B display the ground truth and predictions for 16 test images for the best performing model, YOLOv5m.

4.3. Limitations

Presently, a major limitation of the proposed model is its inability to localize and classify two of the interesting cellular events described in Section 1. A brief

Table 5: Per-category mAP@0.5 on the test set for the 3-class dataset

Approach	Early Mitosis	Late Mitosis	Apoptosis
Faster-RCNN	0.910	0.851	0.870
RetinaNet	0.902	0.855	0.934
YOLOv2	0.958	0.686	0.792
YOLOv5s	0.945	0.852	0.952
YOLOv5m	0.955	**0.882**	0.948
YOLOv5m hyp	**0.964**	0.881	**0.982**
YOLOv5x	0.927	0.860	0.980
YOLOX	0.923	0.780	0.811
VFNet	0.925	0.835	0.938

discussion on the limitations of the model in detecting these two classes is presented below.

Detection of Multipolar events: The current model will not be able to detect Multipolar events, since the multipolar case was removed by the training set due to the cited data imbalance problem. If and when the model encounters an event other than those it was trained on, needless to say it will identify it as another event, probably the one that was present in the training set and is most similar to it in terms of the features extracted by the model. For example, Fig. 10-a) shows a Tripolar division event that the model had never encountered before. This has been identified as Late Mitosis, the class that is morphologically the closet to the mentioned scenario. Similarly, another example in Fig. 10-b) and c) shows the presence of an unknown object (dark spot) captured during the microscopy experiment, which the model detects as Cell Death/Apoptosis in frame c) but correctly recognizes as background in frame b).

Detection of Failure of Division events: Another limitation of the current approach is the inability to recognize one of the key events, Failure of Division. As discussed earlier in Section 3, the event cannot be recognized based only on cell phenotype or spatial information, but requires temporal information. In order to overcome this problem, an approach could be based on cell tracking methods utilizing object-ID association and specific characteristics. A visualization of the event is present in Fig. 11-d). A specific event characteristic is that the cell undergoes "Early Mitosis" state for a quite large period of time and then without any division event, the cells goes back into the background. These specific properties of the said event along with tracked cell ID could be used to identify the cell and detect the event based on the time it remains in the "Early Mitosis" class or by introducing another class of "Interphase" cells present in the background and discover when the cell changes its class, from "Early Mitosis" to "Interphase".

5. Discussion

Among the multiple approaches theorized and applied for event localization and classification, the usage of object detection algorithms by treating the individual event instances as individual objects in each frame was observed to be highly applicable, as is evident from the results. In this section, we will investigate the above results, which led us to the final conclusions.

5.1. 4-class dataset

Several experiments were performed on the complete primary dataset including the Tripolar Division event to observe if the effect of class imbalance could be removed. From the 4-class confusion matrix presented in Fig. 13, it can be seen that the model failed to detect all the Tripolar cases in the test set. This is because the model confidence values for the predictions for this class were very low (0.19 max), so that even a low confidence threshold of 0.25 for the mAP eliminated all the predictions for this class, if there were any. At a lower confidence threshold, the detection on the test dataset did predict Tripolar division cases, but this strongly increased the number of wrongly detected objects. Furthermore, it can also be observed that the model was unable to discriminate between the two very similar classes, Dividing/Late Mitosis and Cell Death/Apoptosis. The high recall and low precision values confirm the results of the system having a large number of results, but the majority of the predicted labels were indeed incorrect.

Hence, based on these inferences, it was decided to omit the minority class in order to improve model performance over the remaining event classes. The images associated with this class were stored for future use, when more class samples will be available.

5.2. 3-class dataset

The omission of the Tripolar class resulted in the 3-class dataset that was further balanced using under- and over-sampling techniques. Subsequent results on this dataset revealed some interesting points discussed below.

The best results obtained with the YOLOv5m model with hyperparameter tuning may be attributed to the model architecture itself, as well as the bag of freebies and specials which use mosaic images in each batch instead of using single images.

The increase in performance in the hyperparameter-tuned model of YOLOv5 was to be expected as hyperparameters are what guide its training. With optimal values, the same model performed better than the initial version. The final list of hyperparameters is listed in Appendix B, in Fig. 16.

An astonishing result reported was the better performance of the YOLOv5 model compared to YOLOX. However, this result is in accordance with the results reported by Keles et al. (2022) in the paper titled "Evaluation of YOLO Models with Sliced Inference for Small Object Detection," where YOLOv5 models surpassed
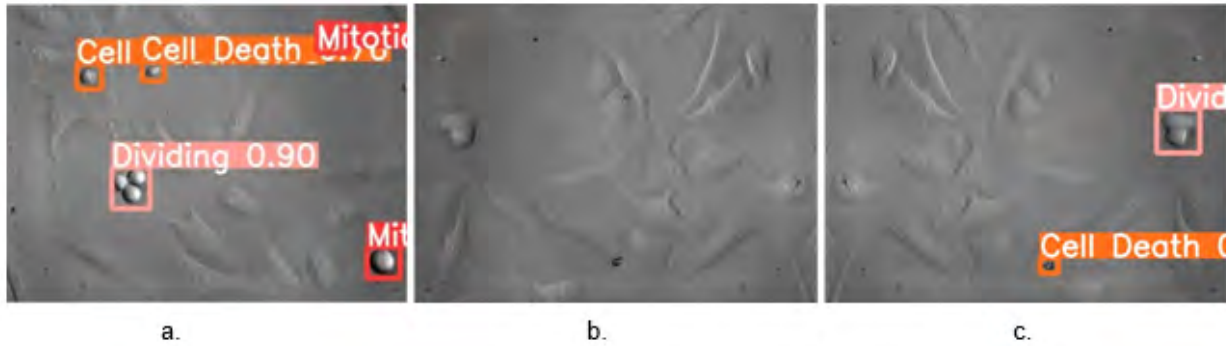
Figure 10: Predictions using the YOLOv5m model on some frame instances: a) with a "Tripolar Division" event detected as Late Mitosis ("Dividing"); b) with a small artifact (dark spot in the bottom center), correctly identified as a background object; c) with the same dark spot artifact as in b) still appearing in a subsequent frame of the sequence, in this case detected as Cell Death.

all the YOLOX models on the VisDrone2019Det dataset [Zhu et al. (2021)].

The performance of YOLOXs was observed to have been severely impacted from the validation dataset to the test dataset where it did not perform well. This might have been a typical example of over-fitting of the data.

Furthermore, one key factor also may be attributed to the reporting of False Positives, which are technically not False positives. There have been some cases where the model could recognize the events one or two frames before the event instances actually materialize. These event instances naturally have not been annotated in the dataset and thus are marked as False Positives, while in truth, they are real events that have been detected. A further investigation is required to discover the actual cause.

The per-class mAP scores reported were as expected, high for the Early Mitosis class and lower and equivalent for the other two classes. This may be due to the fact that the Early Mitotic cells are easily distinguishable in the frames as large rounded objects with smooth contours surrounding them, while the two other classes are possibly confused with each other given their phenotype similarity.

The comparison of the proposed results with the SOTA results in Table 1 cannot be considered a clear and fair method since each of them is obtained on a different dataset. A look limited at the metric values only reveals that the proposed results are comparable to SOTA metric values. However, it should be noted that none of the previously mentioned methods works on the detection of multiple events and that they mostly focus their proposal on one event. For example, Nishimura and Bise (2020) focuses on apoptosis and Su et al. (2017) on the detection of Mitoses. Furthermore, it also should be noted that most of the previous work has been based on Phase contrast microscopy, whereas little to no work has been done in the field for DIC microscopy, which is equally, if not more important, in live-cell imaging.

As discussed in Section 3, one of the main goals of the project is also to look into providing an application of the work and make it more accessible. Event detection is an important problem to tackle not only on U2OS cell lines but also on other cell lines. As the performance of the model would obviously decrease in the case the cells do not appear visually similar, we proposed an easy solution of providing customizable training and testing scripts over Google colab as a mode of application. This concept is exactly taken from the work of Von Chamier et al. (2020) for applications of Deep Learning to Microscopy. This will allow the laboratory personnel to train detection models on their own annotated data, thereby removing the need to rely on algorithms trained on larger datasets.

6. Conclusions

In the present work, we introduce a stratified dataset composed of U2OS cells in DIC time-lapse microscopy videos annotated with bounding boxes for four different cellular events: Early Mitosis, Late Mitosis, Apoptosis, and Multipolar Division. We discussed several different approaches, including spatio-temporal detection using CNN-LSTM and anomaly-based detection, and finally demonstrated that an object detection-based approach to the localization and classification of these events is well-founded. We also demonstrated the use of several different state-of-the-art algorithms on the proposed dataset and were able to provide a google colab-based pipeline.

The proposition made in this study to treat event detection as a multi-class object detection problem and then use the current state-of-the-art methods could be very functional as well as applicable. As discussed, such an approach could also mitigate the issues related to the unavailability of large public datasets and dependence on a single pre-trained model. It allows the individuals to define their own datasets and models and train them to get an application that removes the issue of the

inability of the model to perform well with variability in cell lines as well. We have been able to show through the study that object detection models pre-trained on images other than biomedical images can still provide good results on time-lapse microscopy videos, thus reducing the size of annotated data needed to leverage deep learning models. Object detection in computer vision is still an actively worked-on application, and with new advances in this domain, the analysis of digital microscopy also proceeds forward.

7. Future Work

Event detection in live-cell images is still a subject that is being researched. From the point of view of the proposed work, the future work in the field could be divided into two sections.

- **Downstream Analysis:** An important analytical step in time-lapse microscopy is the calculation of duration of complete events as well as the duration of states in which the cell remains within that event. This could be achieved by tracking the cells throughout their changes in cell states. One approach discussed but not completed was to include an IOU based association or tracking of individual cells as suggested by between different frames. Another approach would be to use deep learning based cell tracking methods. With these tracking data, change in the class of the cell in new frame could be used as a change-point in order to perform various downstream analysis including calculation of event time, and detection of new events such as Failure of cell division.

- **Alternate Method:** An alternate method for detection of cellular events that was looked into but not approached because of time and data limitations is using anomaly detection as a key tool to identify abnormal or anomalous cellular events. Deep learning architechtures such as autoencoders and variational autoencoders have proved to be vital in detection of anomalous events in datasets related to pedestrian, security, and even fraud detection. The same concept could be utilized in order to detect abnormal cellular events in time-lapse videos.

Acknowledgments

References

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C., 2021. Vivit: A video vision transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6836–6846.

Aubreville, M., Stathonikos, N., Bertram, C.A., Klopleisch, R., ter Hoeve, N., Ciompi, F., Wilm, F., Marzahl, C., Donovan, T.A., Maier, A., et al., 2022. Mitosis domain generalization in histopathology images–the midog challenge. arXiv preprint arXiv:2204.03742 .

Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 .

Bodla, N., Singh, B., Chellappa, R., Davis, L., . Improving object detection with one line of code. arxiv 2017. arXiv preprint arXiv:1704.04503 .

Brown, C.M., 2014. Live-cell techniques—advances and challenges. Cell Adhesion & Migration 8, 429–429.

Caldon, C.E., Burgess, A., 2019. Label free, quantitative single-cell fate tracking of time-lapse movies. MethodsX 6, 2468–2475.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C.C., Lin, D., 2019. MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 .

Furcinitti, P., 2013. Methods for preparing differential interference contrast (dic) images for cell tracking and 3-d volume rendering with imagej. Journal of Biomolecular Techniques: JBT 24, S72.

Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J., 2021. Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 .

Grinberg, M., 2018. Flask web development: developing web applications with python. " O'Reilly Media, Inc.".

Huh, S., 2013. Toward an automated system for the analysis of cell behavior: Cellular event detection and cell tracking in time-lapse live cell microscopy. Ph.D. thesis. Carnegie Mellon University.

Jiang, Q., Sudalagunta, P., Meads, M.B., Ahmed, K.T., Rutkowski, T., Shain, K., Silva, A.S., Zhang, W., 2020. An advanced framework for time-lapse microscopy image analysis. bioRxiv .

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., NanoCode012, Kwon, Y., TaoXie, Fang, J., imyhxy, Michael, K., Lorna, V, A., Montes, D., Nadar, J., Laughing, tkianai, yxNONG, Skalski, P., Wang, Z., Hogan, A., Fati, C., Mammana, L., AlexWang1900, Patel, D., Yiwei, D., You, F., Hajek, J., Diaconu, L., Minh, M.T., 2022. ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference. URL: https://doi.org/10.5281/zenodo.6222936, doi:10.5281/zenodo.6222936.

Jung, A.B., Wada, K., Crall, J., Tanaka, S., Graving, J., Reinders, C., Yadav, S., Banerjee, J., Vecsei, G., Kraft, A., Rui, Z., Borovec, J., Vallentin, C., Zhydenko, S., Pfeiffer, K., Cook, B., Fernández, I., De Rainville, F.M., Weng, C.H., Ayala-Acevedo, A., Meudec, R., Laporte, M., et al., 2020. imgaug. https://github.com/aleju/imgaug. Online; accessed 01-Feb-2020.

Keles, M.C., Salmanoglu, B., Guzel, M.S., Gursoy, B., Bostanci, G.E., 2022. Evaluation of yolo models with sliced inference for small object detection. arXiv preprint arXiv:2203.04799 .

La Greca, A.D., Pérez, N., Castañeda, S., Milone, P.M., Scarafía, M.A., Möbbs, A.M., Waisman, A., Moro, L.N., Sevlever, G.E., Luzzani, C.D., et al., 2021. celldeath: A tool for detection of cell death in transmitted light microscopy images by deep learning-based visual recognition. Plos one 16, e0253666.

Lang, W., 1982. Nomarski differential interference-contrast microscopy. Carl Zeiss Oberkochen.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Lu, H., Li, J., Martinez-Paniagua, M.A., Bandey, I.N., Amritkar, A., Singh, H., Mayerich, D., Varadarajan, N., Roysam, B., 2018. Deep learning solution to timing (time-lapse microscopy in nanowell grids). URL: https://github.com/troylhy1991/DEEP-TIMING.

Mao, Y., Yin, Z., 2017. Two-stream bidirectional long short-term memory for mitosis event detection and stage localization in phase-contrast microscopy images, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 56–64.

Moore, B.E., Corso, J.J., 2020. Fiftyone. GitHub. Note: https://github.com/voxel51/fiftyone .

Murphy, D., Spring, K., Parry-Hill, M., Davidson, M., 2017. Comparison of phase contrast and dic microscopy. Nikon Instruments Inc.) https://www. microscopyu. com/tutorials/comparison-of-phase-contrast-and-dic-microscopy 16.

Naso, F.D., Sterbini, V., Crecca, E., Asteriti, I.A., Russo, A.D., Giubettini, M., Cundari, E., Lindon, C., Rosa, A., Guarguaglini, G., 2020. Excess tpx2 interferes with microtubule disassembly and nuclei reformation at mitotic exit. Cells 9, 374.

Nishimura, K., Bise, R., 2020. Spatial-temporal mitosis detection in phase-contrast microscopy via likelihood map estimation by 3dcnn, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE. pp. 1811–1815.

Padilla, R., Passos, W.L., Dias, T.L., Netto, S.L., da Silva, E.A., 2021. A comparative analysis of object detection metrics with a companion open-source toolkit. Electronics 10, 279.

Phan, H.T.H., Kumar, A., Feng, D., Fulham, M., Kim, J., 2019. Semi-supervised estimation of event temporal length for cell event detection. arXiv preprint arXiv:1909.09946 .

Redmon, J., Farhadi, A., 2017. Yolo9000: better, faster, stronger, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7263–7271.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28.

Rosenthal, C.K., 2009. Contrast by interference. Nature Cell Biology 11, S11–S12.

Skalski, P., 2019. Make Sense. https://github.com/SkalskiP/make-sense/.

Su, Y.T., Lu, Y., Chen, M., Liu, A.A., 2017. Spatiotemporal joint mitosis detection using cnn-lstm network in time-lapse phase contrast microscopy images. IEEE Access 5, 18033–18041.

Von Chamier, L., Laine, R.F., Jukkala, J., Spahn, C., Krentzel, D., Nehme, E., Lerche, M., Hernández-Pérez, S., Mattila, P.K., Karinou, E., et al., 2020. Zerocostdl4mic: an open platform to use deep-learning in microscopy. BioRxiv .

Yadav, G., Maheshwari, S., Agarwal, A., 2014. Contrast limited adaptive histogram equalization based enhancement for real time video system, in: 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2392–2397. doi:10.1109/ICACCI.2014.6968381.

Zhang, H., Wang, Y., Dayoub, F., Sunderhauf, N., 2021. Varifocalnet: An iou-aware dense object detector, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8514–8523.

Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., Ling, H., 2021. Detection and tracking meet drones challenge. IEEE Transactions on Pattern Analysis and Machine Intelligence , 1–1doi:10.1109/TPAMI.2021.3119563.

8. Appendix A. Vision Transformer for event classification

An earlier iteration of this research involved the application of previous state-of-the-art adaptations in video classification, such as CNN-LSTM, and newer adaptations, such as Arnab et al. (2021), in order to classify the events while taking into account the spatial and temporal information obtained from the videos. The proposed method included using sequences of single-cell image patches of events as input to train the video classifier. This model would also require a detection or candidate extraction framework, as suggested in Figure 12.

For this purpose, a dataset of single-cell sequences was created from the 24 raw time-lapse microscopy videos where single-cell image sequences of events were cropped out using a 51X51 window such that the image would contain only a single cell at a time. The event sequences were then classified into one of the 4 different event classes: Normal Mitosis, Apoptosis, Multipolar Division, and Failure of Division. The dataset was built from 10 videos encompassing approximately 15 event instances from the four different event classes. Each event instance video contained 10-15 frames. Data augmentation techniques, such as horizontal/vertical flipping and random rotation, were used. The number of epochs and the batch size were set constant at 30 and 2, respectively.
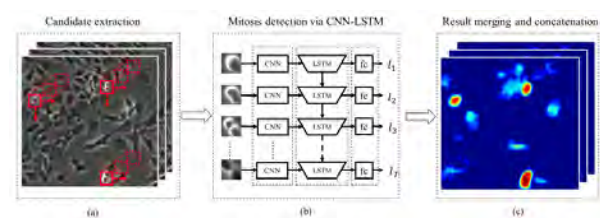
Figure 12: Proposed Network taken from Su et al. (2017)

The dataset was split into ten training event videos containing at least two videos for each event class and five testing videos containing at least one event class each. Spatial data augmentation techniques were then applied on each individual frame, such as horizontal and vertical flipping, and rotations at 45, 135, and 225 were used to increase the dataset size to 5-fold. Two keras-based video classifiers, CNN-LSTM and a CNN-RNN consisting of GRU layers, were trained on the dataset mentioned above. The CNN-LSTM-based model was able to achieve the a better accuracy of 51.36 among the two models on the test dataset.

The low accuracy on our dataset compared to other similar works using CNN-LSTMs may be attributed to
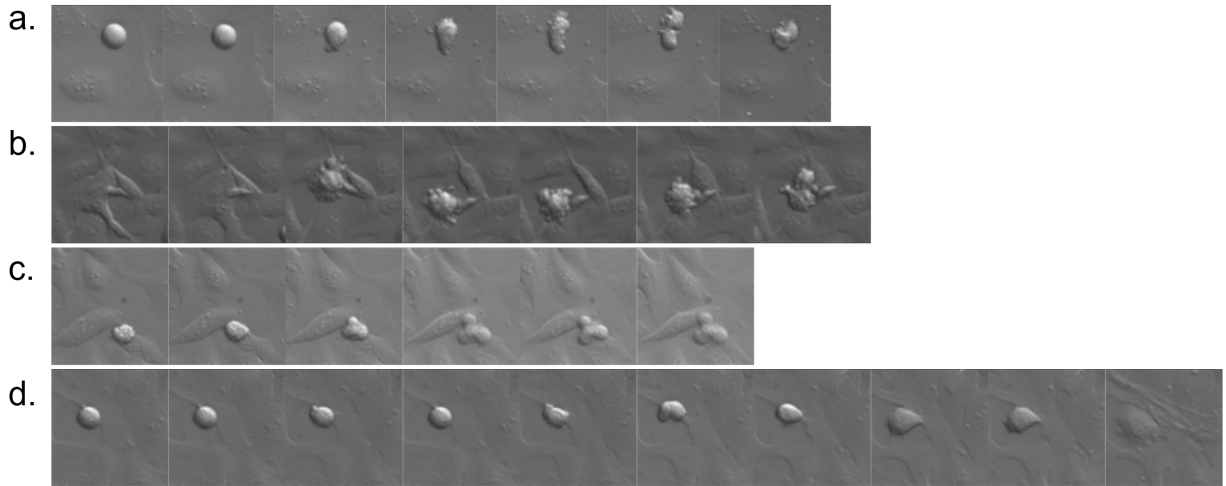
Figure 11: Examples of single-cell patches of size 51X51 extracted from the raw time-lapse videos to build the training dataset for CNN-LSTM/Vision Transformer-based classification. The sequences depict the four different events: a. Normal mitosis, b. Apoptosis, c. Multipolar/Tripolar Division, and d. Failure of Division

Table 6: Performances achieved on the single-cell sequence dataset

Approach	Accuracy	Precision	Recall
CNN-GRU	0.25	0.38	0.402
CNN-LSTM	0.5136	0.66	0.66

the lack of enough data at the time of testing this algorithm. The datasets used by Su et al. (2017) and Mao and Yin (2017) consist of a staggering amount of 2000 and 500 mitotic event sequences, respectively. An effort to include these individual datasets was also made, but the datasets were not publicly available.

Furthermore, to the best of our knowledge, there are currently no available studies that employ a transformers-based video classification approach for event classification in time-lapse microscopy videos. Hence, this approach was discontinued due to time constraints and a lack of enough data in the form of independent videos at the time. With enough data, this approach could have a lot of potential and thus, has been shelved for a future study. Furthermore, the use of vision transformers for video classification could further improve results from previous studies which used CNN-LSTM for the detection of mitosis.

9. Appendix B. Additional Figures Related to YOLOv5
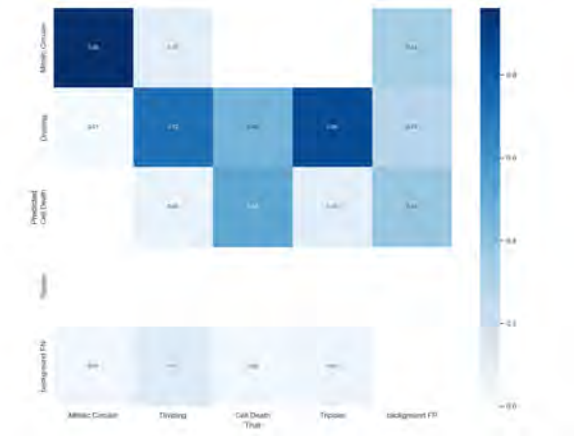
Figure 13: Confusion matrix on the test set for the 4-class dataset

Figure 14: Examples of ground truths for 16 test images.
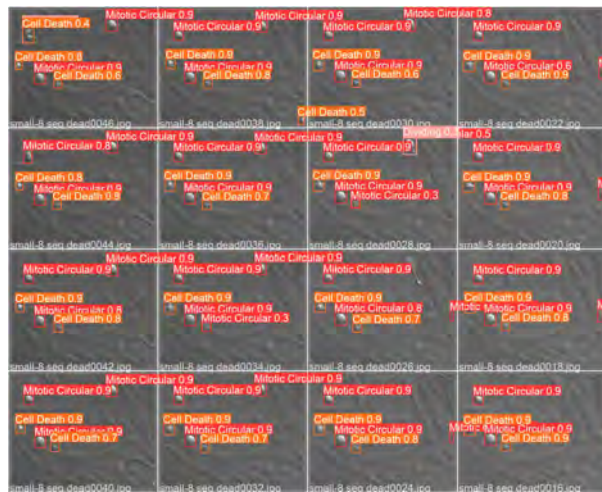
Figure 15: Predictions made using YOLOv5m on the test images of Figure. 14.

Figure 17: Examples of a training input batch with 16 mosaic images with different degrees of contrast.

```
# YOLOv5 Hyperparameter Evolution Results

lr0: 0.01
lrf: 0.01
momentum: 0.937
weight_decay: 0.0005
warmup_epochs: 3.0
warmup_momentum: 0.8
warmup_bias_lr: 0.1
box: 0.05
cls: 0.5
cls_pw: 1.0
obj: 1.0
obj_pw: 1.0
iou_t: 0.2
anchor_t: 4.0
fl_gamma: 0.0
hsv_h: 0.015
hsv_s: 0.7
hsv_v: 0.4
degrees: 0.0
translate: 0.1
scale: 0.5
shear: 0.0
perspective: 0.0
flipud: 0.0
fliplr: 0.5
mosaic: 1.0
mixup: 0.0
copy_paste: 0.0
anchors: 3.0
```

Figure 16: Final hyperparameter file obtained after hyperparameter optimization of the YOLOv5m model.

6.16

Compressing U-Net inspired Transformer based Segmentation Models using Information Flow

Syed Nouman Hasany[a], Caroline Petitjean[b], Fabrice Mériaudeau[c]

[a]*syednoumanhasany1997@gmail.com*
[b]*caroline.petitjean@univ-rouen.fr*
[c]*fabrice.meriaudeau@u-bourgogne.fr*

Abstract

Transformer models have recently started gaining popularity in Computer Vision related tasks. Within Medical Image Segmentation, segmentation models such as TransUNet have incorporated transformer blocks alongside convolutional blocks while remaining faithful to the popular U-Net architecture. The rationale behind this is to supplement the local information obtained from convolutional kernels with the global information obtained from transformer blocks. The present work examines information flow within transformer blocks of three such segmentation models: (i) TransUNet, (ii) 2D CATS, and (iii) 2D UNETR. An analysis of attention information reveals that the transformer blocks for the TransUNet effectively achieve a global receptive field starting from the first block, whereas the 2D CATS and 2D UNETR achieve it starting from the third block. Based on the analysis, compressed versions of these models are proposed in which all blocks after the first such block which effectively achieves a global receptive field are discarded helping reduce the number of parameters to around 40% of the original parameters for 2D CATS and around 25% for TransUNet and 2D UNETR. With the help of three different datasets (IBSR 18, EMIDEC, Synapse multi-organ), it is shown that compression can reduce model parameters without seriously sacrificing model performance. For the IBSR 18, the dice metric drops by a maximum of 0.21% for the 2D UNETR - Compressed, for the EMIDEC dataset it drops by a maximum of 2.16% for the 2D UNETR - Compressed, and for the Synapse multi-organ dataset it drops by a maximum of 2.65% for the 2D CATS - Compressed.

Keywords: vision transformer, segmentation, attention maps, model compression

1. Introduction

Following the success of AlexNet (Krizhevsky et al., 2017), the last decade of Medical Image Analysis has been dominated by Convolutional Neural Networks (CNNs). Recently, however, an alternate approach towards image analysis has been proposed which utilizes transformer blocks.

Transformer blocks differ significantly from convolutional blocks in that they can theoretically learn global relationships within an image. Convolutional blocks, on the other hand, can only extract local information. This difference primarily stems from the fact that a convolutional block effectively has a limited field of view (i.e. receptive field) whereas for a transformer block, the field of view is essentially the entire image.

Similar to AlexNet, the transformers blocks were popularized within computer vision in the context of image classification with the proposal of the Vision Transformer (ViT) model (Dosovitskiy et al., 2021). Naturally, the application domain soon broadened and transformers were applied to other image analysis tasks as well. The present study is concerned with one such task being that of medical image segmentation. Image segmentation is the task of assigning a label to each pixel of an image. In contrast to image classification which generally produces a single label as the output for each image, image segmentation is mostly defined such that it results in as many labels as there are pixels in the image.

Unlike image classification where the ViT relied solely on transformer blocks, image segmentation models incorporating transformer blocks have also tended

to incorporate convolutional blocks. As far as medical image segmentation is concerned, the encoder-decoder based U-Net architecture (Ronneberger et al., 2015) and its variants have been the relied upon workhorse. This trend seems to be reflected even in segmentation models which incorporate transformer blocks. A number of such models have maintained the basic encoder-decoder U-shaped architecture. The present study, too, is concerned with this particular family of segmentation models i.e. U-Net inspired segmentation models which have incorporated transformer blocks. Three such architectures - each of which modify the standard U-Net architecture differently - are considered here:

- TransUNet (Chen et al., 2021)

- CATS (Li et al., 2022)

- UNETR (Hatamizadeh et al., 2022b)

For architectures belonging to this family it is generally argued that the idea behind adding transformer blocks is to utilize the transformer's strength of extracting global relations and to complement it with the convolution's strength of extracting local relations (Chen et al., 2021). The present study investigates this claim by analyzing how information flows in these models. Specifically, we can narrow in on the information flow contributed from the transformer blocks. This allows us to examine the veracity of the aforementioned philosophy as to whether convolutional blocks and transformer blocks indeed contribute differently to the overall information flow. As such there are three key ideas behind the following work. The first is to utilize U-Net inspired segmentation models incorporating transformer blocks and evaluate them on three medical imaging datasets:

- Synapse multi-organ dataset

- EMIDEC dataset

- IBSR 18 dataset

The second idea is to work with model interpretability techniques and investigate the contribution of transformer blocks towards information flow in such architectures. The third idea is to utilize results from this investigation in order to simplify existing architectures.

2. State of the art

Image classification was dominated by CNNs in the last decade based on the success of architectures such as AlexNet (Krizhevsky et al., 2017), ResNet (He et al., 2016), and EfficientNet (Tan and Le, 2019). Inspired from Natural Language Processing (NLP), however, a transformer based model having no convolutional blocks was proposed in 2020. This was the Vision Transformer model (ViT) (Dosovitskiy et al., 2021)

which divided an image into patches. Embeddings extracted from each patch - similar to word token embeddings in NLP - were passed on to a series of transformer blocks before adding a multilayer perceptron to the final layer for a classification decision. The same paper also proposed a hybrid model in which instead of the transformer blocks directly operating on the original image, they were applied to a feature representation of the original image obtained from a CNN based backbone. Since they lack the inductive bias already present in CNNs, transformer models have been known to require a large amount of data. Vision transformer models were pretrained on JFT-300M dataset, and the final models are available online for transfer learning.

Similar to image classification, image segmentation was also dominated by CNNs. The encoder-decoder based U-Net (Ronneberger et al., 2015) was a relatively popular model inspiring derivatives such as V-Net (Milletari et al., 2016) and U-Net++ (Zhou et al., 2018). Attention was also utilized in some derivatives such as the Attention U-Net (Oktay et al., 2018), Attention Unet++ (Li et al., 2020), and Attention Gated Network (Schlemper et al., 2019). The earliest incorporation of transformer blocks, however, was done in TransUNet in 2021 (Chen et al., 2021). The idea behind TransUNet was to replace the bottleneck convolutional layer of a U-Net (with a ResNet backbone as encoder) with transformer blocks.

Following TransUNet, other segmentation models incorporating transformer blocks were also proposed such as the TransBTS (Wang et al., 2021) which expanded upon TransUNet to be directly applicable to 3D images, LeViT-UNet (Xu et al., 2021) which replaced the ViT transformer in TransUNet with LeViT, CoTr (Xie et al., 2021) in which the transformer blocks were applied not only to the bottleneck layer but to the remaining layers of the multi-scale feature map as well, Swin UNETR (Hatamizadeh et al., 2022a) in which the U-Net encoder was replaced by Swin transformer blocks, etc.

Many of the proposed models follow the encoder-decoder based U-Net architecture and incorporate transformer blocks on the encoder side. While TransUNet replaces the bottleneck layer with transformer blocks, other models such as CATS (Li et al., 2022) introduce a transformer based parallel path, the information from which is fused with that coming from the convolutional path before getting passed on to the decoder. Another model is the UNETR (Hatamizadeh et al., 2022b) which gets rid of all convolutional blocks within the encoder - barring one - and replaces them with transformer blocks.

In addition to solving computer vision tasks such as classification and segmentation, another important field of research has been that of model interpretability. Model interpretability often involves examining the flow of information in order to understand how a model arrived at its eventual decision. For CNNs, multiple

such techniques exist such as Class Activation Mapping (CAM) (Zhou et al., 2016) which is based on obtaining a linear combination of the activation maps given the output category one wishes to inspect. CAM gave rise to multiple techniques such as Grad-CAM (Selvaraju et al., 2020) which utilizes gradient information in order to find the linear combination coefficients and Ablation-CAM (Desai and Ramaswamy, 2020) which utilizes masking of feature maps in order to find the linear combination coefficients.

For vision transformer models, information flow often involves the inspection of attention values. Due to the nascency of vision transformers there are relatively few techniques which exist in this domain. (Samira and Willem, 2020) proposed two such techniques, one being Attention Rollout which is based on recursive matrix multiplication, and the other being Attention Flow which is based on the maximum flow problem from graph theory. Both of these techniques attempt at quantifying the propagation of attention values from earlier blocks to the later blocks. Attention Rollout was also utilized in the vision transformer paper (Dosovitskiy et al., 2021) in order to interpret the results. Other techniques have also been proposed such as the one by (Chefer et al., 2021), which utilizes relevance propagation and instead of just focusing on transformer blocks, takes the entire network into account.

3. Material and methods

3.1. Transformer

The transformer model was initially proposed in the context of machine translation (Vaswani et al., 2017). The self-attention based transformer model can essentially be thought of as a representation learning mechanism for sequential data. With every successive transformer block, the representation of individual sequence units is modified taking all other sequence units into account. This is in stark contrast to the Recurrent Neural Networks or the 1D Convolutional Networks which only explicitly focus on a neighborhood around a particular sequence unit in order to modify its existing representation.

An individual transformer block generally consists of two layers, a self-attention layer and a multilayer perceptron. The input to the transformer block is an input sequence of length N, with each unit of the sequence being represented by an embedding of size D. In order to inject positional information into the model, a positional embedding is added to each input embedding. These positional embeddings can either be pre-determined or learnt during the training process. For each input embedding, three vectors of size d_k representing the "key", "query", and the "value" are obtained via a simple matrix multiplication involving the input embedding and the key, query, and value matrices respectively. These

matrices are learnt during the training process. For each input unit, it is determined as to which input units (including itself) should contribute towards its next representation. This is achieved by taking a dot product between the query vector of the concerned input unit and the key vector of all units in the input sequence. A softmax is then applied to a scaled version of this dot product representing the importance of each unit towards the unit in question. The representation is then formed using a linear combination of value vectors such that the results of the softmax form the coefficients of the linear combination. This completes the self-attention mechanism. Self-attention can be applied in a single step for the entire sequence as expressed in the following equation:

$$Attention(Q, K, V) = softmax(\frac{Q \cdot K^T}{\sqrt{d_k}}) \cdot V \quad (1)$$

Where Q, K, V represent the query, key, and value representations of the entire sequence, each of shape $N \times d_k$.

In order to allow for multiple useful representations, instead of using a single self-attention mechanism, a multi-head self-attention mechanism is utilized. Put simply, self-attention is repeated multiple times for each unit, and the eventual representations are concatenated to obtain a final representation. Following self-attention, the representations are passed through a multilayer perceptron whose weights are shared between all units of the sequence.

In addition to the two layers, each transformer block also makes use of layer normalization and residual connections. Both self-attention, and multilayer perceptron are preceded with layer normalization, and succeeded with residual connections. The entire workflow of a self-attention based transformer block (Figure. 1) can, thus, be expressed in the following set of equations:

$$z_l' = MSA(LN(z_{l-1}) + z_{l-1} \quad (2)$$

$$z_l = MLP(LN(z_l')) + z_l' \quad (3)$$

Where MSA refers to multi-head self-attention, MLP refers to multilayer perceptron, z_{l-1} represents the representations from the preceding block, and z_l represents the representations from the current block.

3.2. Vision Transformer

Following the success of transformers in the field of Natural Language Processing, (Dosovitskiy et al., 2021) successfully applied them in the field of Computer Vision, particularly image classification. Since images, unlike textual data, are not inherently sequential, the authors proposed two simple ways in which image data can be made compatible with transformers:
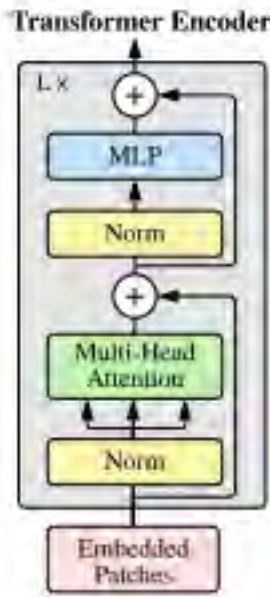
Figure 1: Input representations pass through layer normalization, multi-head self attention, residual connection, a second layer normalization, multilayer perceptron, and a second residual connection before being passed on to the next transformer block (the total being *L*) (Dosovitskiy et al., 2021)

- Raw image patches

- Hybrid architectures

3.2.1. Raw Image Patches

A straightforward approach would be to simply consider each image pixel as an individual input unit to the transformer block. However, owing to the squared computational cost within the self-attention block, this is not feasible. Consequently, the authors propose splitting the image into non-overlapping image patches. Each patch is then flattened, and passed through an embedding layer. The sequence of these input embeddings is what forms the input to the transformer model. Conventionally, for an image of size 224×224, the patch size is taken to be 16×16 leading to 14×14 patches which form 196 input tokens.

In addition to this, since the model is also required to make a classification decision, an extra token is added in the beginning of the sequence - referred to as the "CLS" token. With each transformer block the representation of the input embeddings keep getting modified, and from the last transformer block, the representation corresponding to the "CLS" position is passed through a simple multilayer perceptron followed by a softmax to get a classification decision.

A pictorial representation of the process can be seen in Figure. 2.
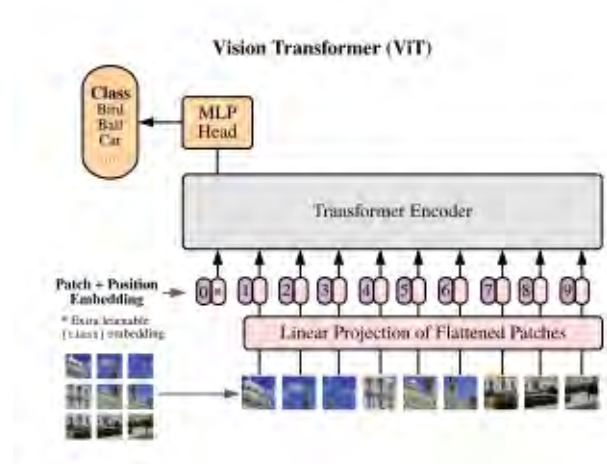
Figure 2: Non-overlapping patches of size 14×14 are extracted from the image which are then flattened and linearly projected. Positional embeddings are then added to each representation, and an extra "CLS" vector is added in the beginning before being passed on to the transformer blocks. Output representations corresponding to the "CLS" location are then fed to a multilayer perceptron from which the final classification output is extracted (Dosovitskiy et al., 2021)

3.2.2. Hybrid Architectures

An alternative to raw image patches is to avoid applying the transformer model directly to the input image. Instead, the authors propose passing the input image through a Convolutional Neural Network, and the transformer model can then be applied to the output of an intermediate convolutional layer. The authors utilize the ResNet family of architectures as their backbone Convolutional Network. The rationale behind this method can be two-fold. Firstly, the spatial dimensions of the intermediate feature maps would have decreased considerably by then, allowing the transformer to be applied such that each pixel position be considered as an individual input. Secondly, this allows for the convolutional network to provide the transformer with an already rich feature representation as its input.

3.3. U-Net inspired Segmentation Models incorporating Transformers

ViT's success has led to the application of transformer models into domains other than image classification, a prime example being that of Image Segmentation. In recent years, Medical image segmentation has been dominated by the famous U-Net architecture (Ronneberger et al., 2015) [Figure. 3] which has, since then, also inspired multiple successful derivatives. U-Net is essentially an encoder-decoder based fully convolutional architecture. In the encoding path the image passes through a series of convolutions and max poolings which successively reduces its dimensions. Once the dimensions have been satisfactorily reduced, the intermediate output is passed to the decoding path where it passes through a series of upsamplings and conven-
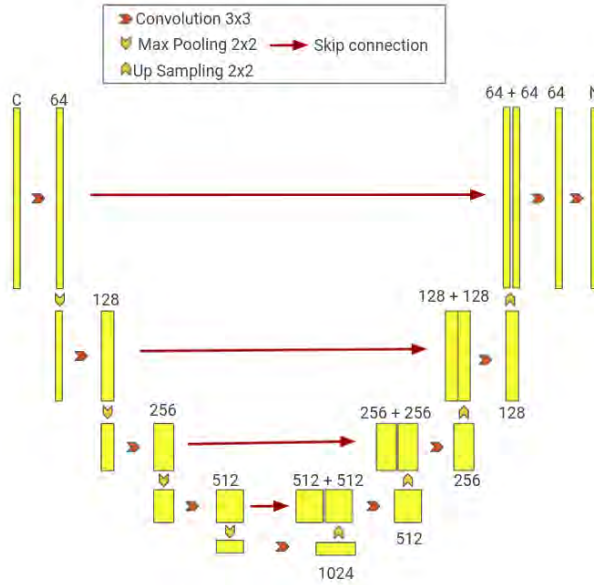
Figure 3: U-Net architecture - the encoder side processes the image using convolutional blocks followed by max-pooling blocks, the decoder side upsamples the output from the encoder bottleneck, concatenates it with the corresponding encoder output and repeats the process until a final convolution is applied corresponding to the number of classes *N* to segment. The numbers in the figure represent the channels from each stage of the process

Figure 4: TransUNet architecture - the encoder side - a ResNet-50 based backbone - processes the image gradually decreasing its spatial dimensions. Once the spatial dimensions reach 14×14, the features are processed by a transformer, the output of which is passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. The numbers in the figure represent the channels from each stage of the process

tional convolutions which successively increase its dimensions. The idea of the encoding path is to construct a global representation of the image whereas the idea of the decoding path is to generate the final segmentation map using that representation. However, since the global representation lacks the local information, U-Net adds skip connections between corresponding downsampling and up- sampling layers so as to include the local information as well as the global information in order to produce the final output.

Interestingly, many transformer based image segmentation models also seem to be inspired by the U-Net architecture, and closely follow the encoder-decoder based construction supplemented with skip connections. Presently, the focus will be on three such architectures each of which modify the encoder portion of the U-Net in a different manner:

- TransUNet - replaces the bottleneck convolutions of the encoder with transformer blocks

- CATS - runs the image through both transformer blocks and convolutions separately and combines the information at each encoder step

- UNETR - replaces the convolutional part of the encoder entirely with transformer blocks barring one convolutional layer

3.3.1. TransUNet

Introduced in 2021, TransUNet (Chen et al., 2021) was one of the first segmentation models which utilized vision transformers. The idea behind TransUNet was to replace the bottleneck portion of the encoder part with a transformer model. In order to utilize available pre-trained architectures, TransUNet makes use of a hybrid vision transformer model introduced in (Dosovitskiy et al., 2021). Assuming a 224×224 sized image, the encoder part consists of feature maps obtained from three blocks of a ResNet based backbone having spatial dimensions of 112×112, 56×56, and 28×28. Following that, the output from the ResNet block has spatial dimensions of 14×14 which are then fed to a transformer block as a sequence of length 196. There are a total of 12 transformer blocks, the output of each of which is 196×768 where 196 is the sequence length, and 768 is the representation dimension for each sequence unit. The output of the final transformer block is then reshaped back to $14 \times 14 \times 768$, followed by a convolutional layer, and an upsampling layer. This is concatenated with the corresponding feature map from the ResNet backbone, and the process continues until the desired image dimensions are reached as can be seen in Figure. 4. It is worth noting that a major difference between this transformer model and the transformer models used in image classification is the lack of an extra input token in the beginning as we are not interested in a classification decision, but are only interested in utilizing the representations obtained from the transformer.

3.3.2. CATS
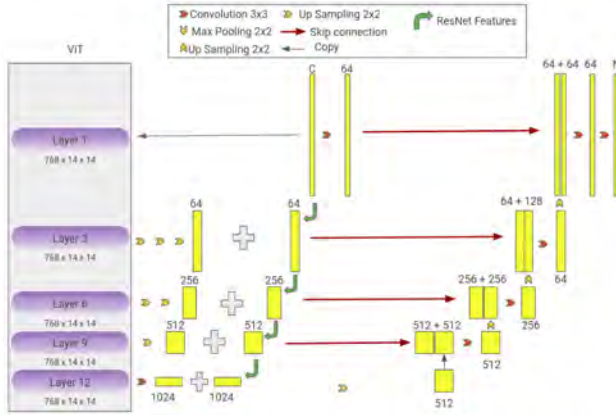
Figure 5: 2D CATS - the encoder side is composed of two parallel pathways. One is composed of a convolutional backbone - a ResNet-50 in this figure - which processes the image by gradually decreasing its spatial dimensions. The other one is composed of a pre-trained ViT which works on 196 (14×14) non-overlapping patches (size 16×16) from the original image and the output from each of its blocks can be reshaped to a spatial dimension of 14×14. The output from the third, sixth, and ninth transformer blocks are passed through a series of upsamplings before being added to the corresponding convolutional backbone output. The output from the twelfth block is passed through a convolutional block, added to the corresponding backbone convolutional output and passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. There is also an independent convolutional branch connecting the image directly to the final decoder layer. The numbers in the figure represent the channels from each stage of the process

Introduced in 2022, CATS (Li et al., 2022) retains the original convolution based U-Net encoder. However, it adds an additional path in which the original image is passed through a transformer model. The information from the convolutional pathway and the transformer pathway is then fused using simple addition before flowing on to the decoder part. While the original CATS paper proposed a network for 3D images, the present work presents a slightly modified version which is supposed to work for 2D images. Furthermore, the CATS architecture utilized a convolutional encoder branch similar to the one proposed in the original U-Net whereas in the modified architecture, most convolutional backbones such as the ones available in PyTorch image models library (Wightman, 2019) can be utilized. Lastly, the CATS architecture does not use a pre-trained transformer model whereas the proposed architecture utilizes a pre-trained ViT.

Assuming an original image of spatial dimensions 224×224, the image is passed through a convolutional pathway and a transformer based pathway separately. The convolutional pathway can consist of a conventional U-Net based encoder or pre-trained feature extractors belonging to architectural families such as ResNet, DenseNet (Huang et al., 2017), EfficientNet, etc. For the present example, assuming ResNet-

50 as the convolutional backbone, the spatial dimension goes from 224×224 to 112×112, 56×56, 28×28, and 14×14. For the transformer pathway, this work assumes a pre-trained ViT-B/16 architecture consisting of 12 transformer blocks. The representation obtained from each block is a sequence of length 196 with each input token having an embedding size of 768. The representation from the third block is reshaped to a spatial dimension of 14×14 followed by three upsampling operations and a convolutional block to bring the spatial dimension to 112×112. This is then added to the corresponding ResNet-50 representation. Similarly, representation from the sixth block is reshaped, and followed by two upsampling operations and a convolutional block before being added to the corresponding ResNet-50 representation. Representations from the ninth and twelfth block follow a similar pattern, except that the twelfth block needs no upsampling operation as its spatial dimensions are already 14×14. In addition to these two pathways, there is also a convolutional layer which connects the input image directly to the final decoding step via a skip connection. The detailed architecture can be seen in Figure. 5.

3.3.3. UNETR
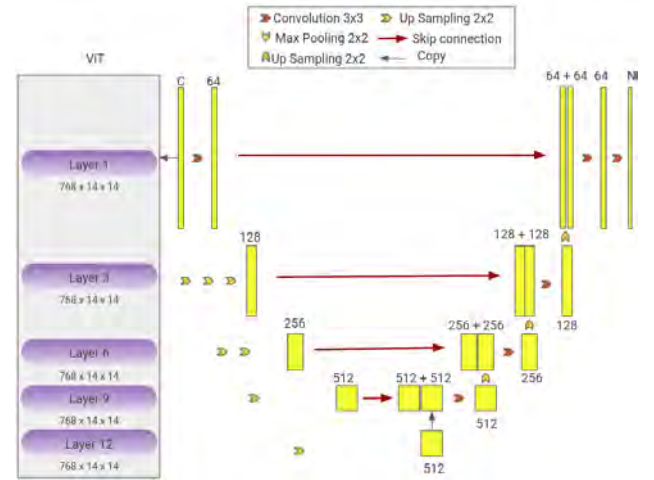
Figure 6: 2D UNETR - the encoder side is composed of a pre-trained ViT which works on 196 (14×14) non-overlapping patches (size 16×16) from the original image and the output from each of its blocks can be reshaped to a spatial dimension of 14×14. The output from the third, sixth, ninth, and twelfth transformer blocks are passed through a series of upsamplings. The upsampled output from the twelfth block is passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. There is also an independent branch connecting the image directly to the final decoder layer. The numbers in the figure represent the channels from each stage of the process

Introduced in 2021, UNETR (Hatamizadeh et al., 2022b) attempts to make an encoder almost entirely based on transformer blocks. Similar to CATS, in the present work the original architecture has been modified such that it works for 2D images instead of 3D, and

it utilizes a pre-trained ViT model instead of randomly initializing the transformer blocks.

Assuming an original image of spatial dimensions 224×224, the image is passed through a pre-trained ViT-B/16 architecture consisting of 12 transformer blocks. Similar to CATS, the representations from the third, sixth, ninth, and twelfth block are reshaped, upsampled, and passed through convolutional blocks. The representations obtained from the encoder pathway end-up having spatial dimensions of 112×112, 56×56, 28×28, and 28×28. Additionally, the input image is also passed through a convolutional layer connecting it to the final decoding step via a skip connection. The detailed architecture can be seen in Figure. 6.
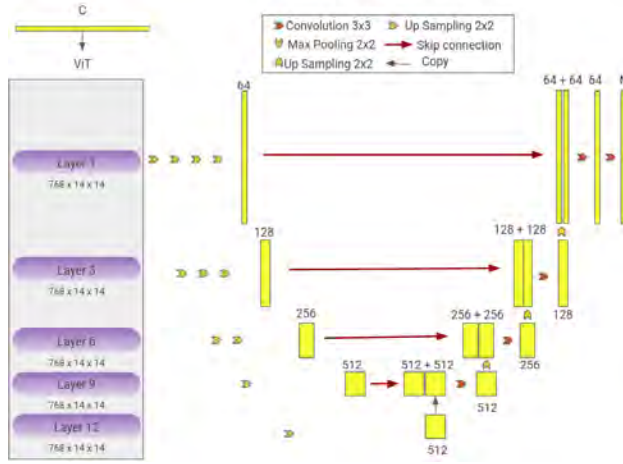
Figure 7: 2D all-trans-UNETR - the encoder side is composed of a pre-trained ViT which works on 196 (14×14) non-overlapping patches (size 16×16) from the original image and the output from each of its blocks can be reshaped to a spatial dimension of 14×14. The output from the first, third, sixth, ninth, and twelfth transformer blocks are passed through a series of upsamplings. The upsampled output from the twelfth block is passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. There is no independent convolutional branch operating on the image. The numbers in the figure represent the channels from each stage of the process

Since UNETR has an encoder which does make use of a single convolutional layer, another architecture is also experimented with which removes that convolutional layer and instead utilizes the reshaped representations from the first transformer block followed by four upsampling operations and a convolutional block. The detailed architecture can be seen in Figure. 7. This modified architecture represents a U-Net based segmentation model where the encoder is entirely transformer based.

3.4. Information Flow

Unlike classical machine learning models which can generally be implemented in a single layer, Deep Learning models are usually many layers deep. Analyzing the flow of information in such models is often useful to determine what kind of information is being learnt in each layer of the model. This is often necessary for interpretability but can also be useful if one wishes to modify the model architecture.

For transformers, instead of focusing on activation maps, it is often more useful to look at the attention values in order to know where attention was being paid in a particular transformer block. The present work considers two approaches for this task. A relatively simple approach is to observe the raw attention values from each block whereas another approach as proposed by (Samira and Willem, 2020) attempts to trace attention values as they propagate from one transformer block to the next.

3.4.1. Raw Attention Values

Transformer models afford us the possibility to not just observe feature maps, but to also observe how much each patch in any block was attending to the other patches (including itself). This can be achieved by visualizing the raw attention values obtained from the dot-product of key and query matrices.

3.4.2. Rollout

While raw attention values can be used to analyze attention within a particular block, (Samira and Willem, 2020) proposed a technique which can be used to infer how attention values propagate from one block to the next. This can essentially be utilized to observe how much attention a patch from the original image is being paid to in a later transformer block. The idea is to simply multiply the attention matrices recursively starting from an earlier block and going towards a later block. In order to take the residual connections within a transformer model into account, an identity matrix is added at each step followed by an averaging of the two. Mathematically, this can be represented as follows:

$$A = \frac{W_{att} + I}{2} \tag{4}$$

$$A(\tilde{l_i}) = \begin{cases} A(l_i)A(\tilde{l}_{i-1}) & i > j \\ A(l_i) & i = j \end{cases} \tag{5}$$

Where $A(\tilde{l_i})$ is the quantification of the attention rollout, and $A(l_i)$ refers to the averaged identity and attention matrices for a particular block. j is taken to be 0.

For the present work, the final result is slightly modified by suppressing the rollout matrix diagonal which would otherwise overshadow the other values in the matrix.

3.5. Datasets

This work experimented using three different medical image segmentation datasets. The first one is the IBSR

18 dataset[1]. It contains the T1-weighted brain MRI images for 15 patients. 10 patients were used for training and 5 for validation leading to 1280 2D slices for training and 640 for validation. The labels are Cerebrospinal Fluid (CSF), Gray Matter (GM), and White Matter (WM).

The second dataset is the Synapse multi-organ dataset[2]. 30 cases of abdominal CT scans are provided, 18 of which are used for training and 12 for validation leading to 2211 2D slices for training and 1568 for validation. The labels are Aorta, Gallbladder, Spleen, Left Kidney, Right Kidney, Liver, Pancreas, and Stomach.

The third dataset is the EMIDEC Challenge dataset (Lalande et al., 2020)[3]. 100 cases of delayed-enhancement cardiac MRI are provided, 80 of which are used for training and 20 for validation leading to 558 2D slices for training and 180 for validation. The labels are Myocardium, Infarction, and NoReflow.

3.6. Pre-processing

The datasets were normalized with each image volume ending up with zero mean and unity standard deviation. For the Synapse multi-organ dataset, dataset is obtained from the TransUNet authors in which, prior to normalizing, the images were clipped within a range of -125 and 275. For the EMIDEC dataset, center cropping was performed with a size of 96×96. Eventually, each slice from all images is resized to 224×224.

3.7. Training Configuration

All models were trained with an AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of $1e - 04$ without any weight decay. For the IBSR 18 and Synapse multi-organ datasets, 80 epochs were utilized whereas for the EMIDEC dataset, 100 training epochs were utilized. In each case, the batch size was 12. The loss function in each case was an average of dice loss and cross-entropy loss.

3.8. Analyzing Information Flow

3.8.1. Attention Information

In order to analyze attention flow for each model, raw attention maps were visualized for all transformer blocks. In general, at each step there is a sequence of length 196, and each unit of that sequence pays attention to each other unit implying 196 attention values per unit. For every unit, an attention map of size 14×14 can be obtained, and since there are 196 units, each of these 14×14 attentions maps can be displayed on a 14×14 grid. In each case raw attention maps from the first, second, third, and sixth transformer block are visualized.

---

[1] http://www.nitrc.org/projects/ibsr
[2] https://www.synapse.org/#!Synapse:syn3193805/wiki/217789
[3] http://emidec.com/dataset

An analysis of raw attention maps for TransUNet (Figure. 8) reveals that, generally, starting from the first transformer block, patches start paying attention to the remaining patches irrespective of their spatial proximity to those patches. For example, for a shape of 14×14, position $(1, 1)$ can potentially pay attention to a patch at position $(14, 14)$. This trend continues for the remaining transformer blocks as well.

Analyzing the raw attention maps from the 2D UNETR (Figure. 9) reveals an interesting pattern. Unlike the TransUNet, in the first block, patches do not pay attention to the remaining patches. In fact, in the first block each patch mostly seems to be paying attention only to itself. For the second block, attention maps reveal that each patch is mostly paying attention to patches which are within close proximity. This behaviour strongly resembles that of a convolutional kernel passing over an image. From the third block onward, however, each patch seems to be paying attention to other patches without any preference to spatial proximity. For the 2D UNETR which is purely based on a transformer encoder and has had its convolutional branch removed, the behaviour is similar (Figure. 10).

The 2D CATS architecture also behaves similar to the 2D UNETR (Figure. 11). In the first block, each patch mostly pays attention to itself, in the second block each patch attends to patches within close proximity, and starting from the third block, patches start paying attention without particular concern for spatial proximity.

In addition to raw attention maps, attention rollout is also visualized on a 14×14 grid with each unit on the grid corresponding to a 14×14 attention rollout information. Unlike raw attention maps, only rollout from the first three blocks is visualized.

The conclusions are similar to the ones drawn from raw attention values. For the TransUNet (Figure. 12, attention propagation displays no concern for spatial proximity with respect to the transformer block. For the 2D UNETR (Figure. 13 and the 2D CATS (Figure. 14), attention propagates from the first to the second block with a strong emphasis on local proximity whereas in the third block (and onward), spatial proximity plays no particular role.

3.9. Model Compression

An analysis of attention maps as well as attention rollout indicate that the architectures under consideration might be simplified by departing from the conventional U-Net style. In a conventional U-Net, an encoder contains multiple convolutional blocks primarily because as one goes from one block to the next, the receptive field of the model increases and hence more context can be incorporated. If one contrasts it with the attention information obtained from the 2D UNETR and 2D CAT models, it can be seen that starting from the third transformer block, attention maps already seem to

Figure 8: TransUNet - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from left to right and top to bottom. Maps for the first (top-left), second (top-right), third (bottom-left), and sixth (bottom-right) transformer blocks are plotted

Figure 9: 2D UNETR - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from left to right and top to bottom. Maps for the first (top-left), second (top-right), third (bottom-left), and sixth (bottom-right) transformer blocks are plotted
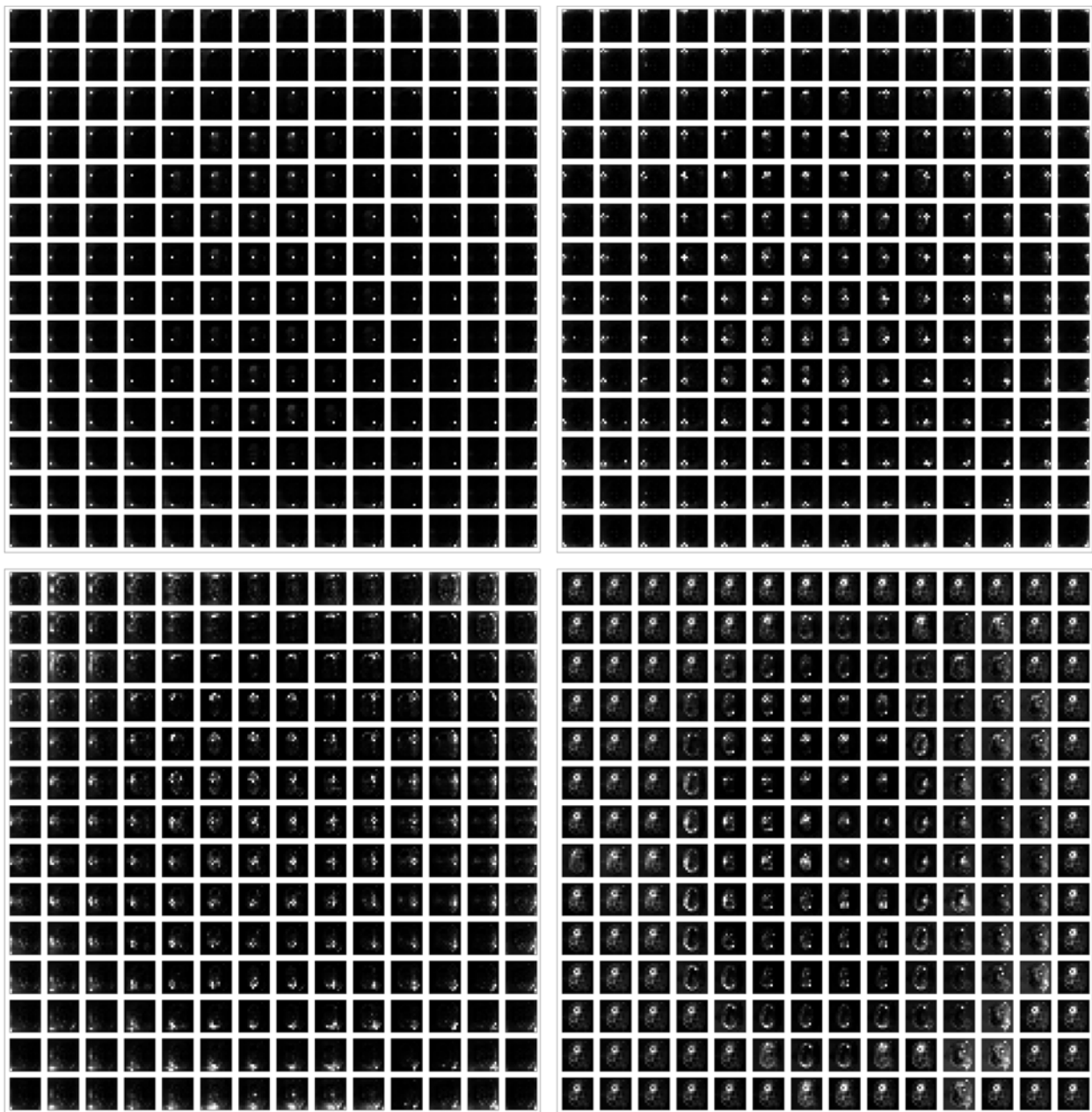
Figure 10: 2D UNETR - all trans - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from left to right and top to bottom. Maps for the first (top-left), second (top-right), third (bottom-left), and sixth (bottom-right) transformer blocks are plotted

Figure 11: 2D CATS - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from left to right and top to bottom. Maps for the first (top-left), second (top-right), third (bottom-left), and sixth (bottom-right) transformer blocks are plotted
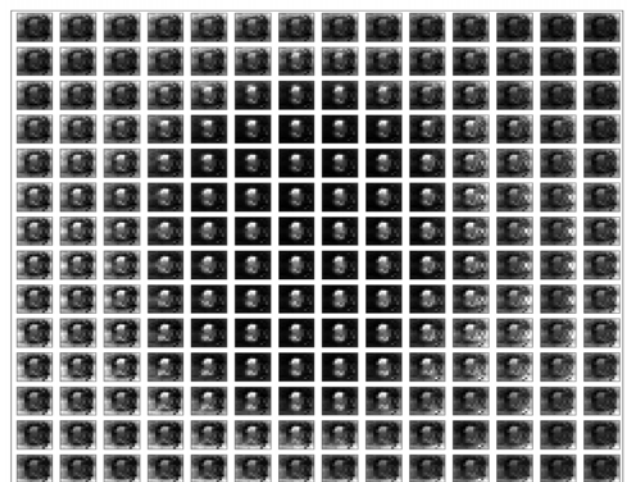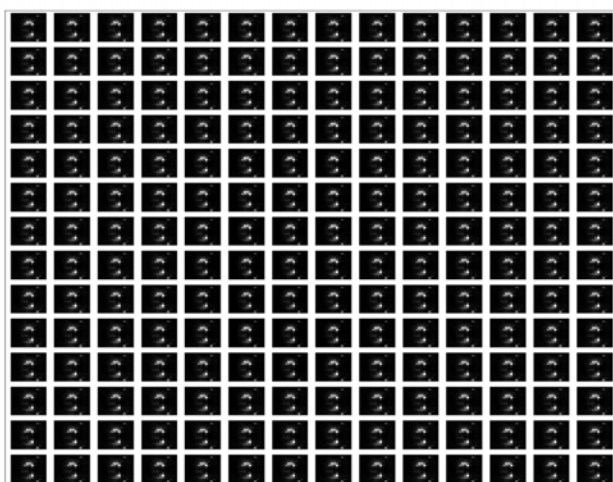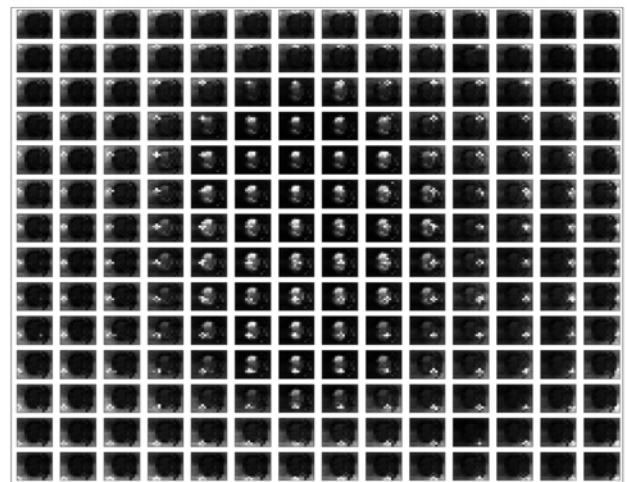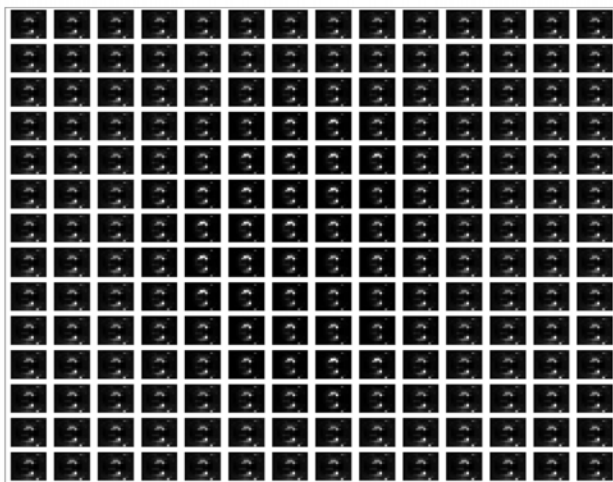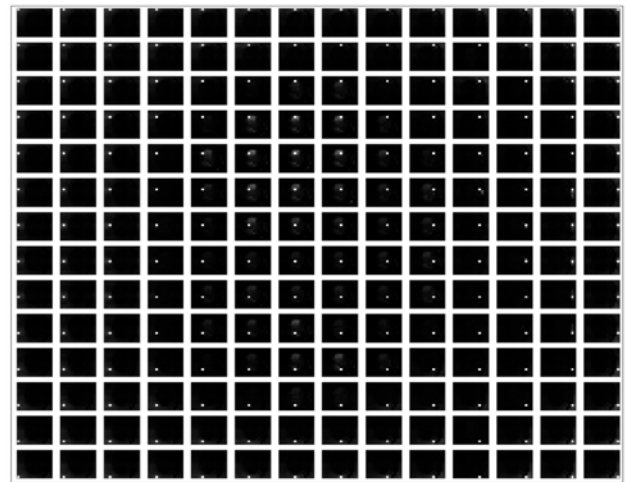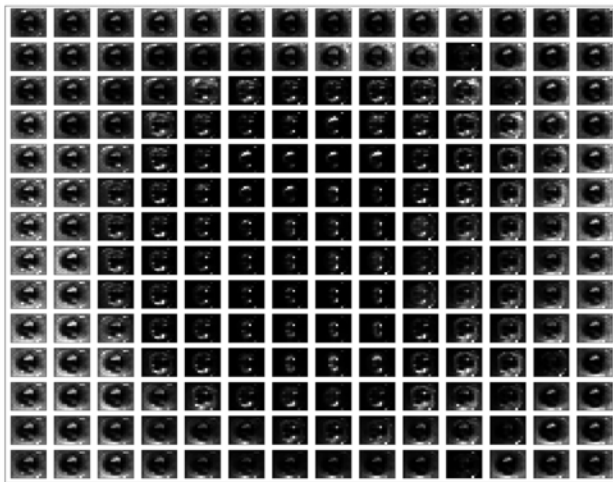
Figure 12: TransUNet - attention rollout for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Rollout for the first (top), second (middle), and third (bottom) transformer blocks is plotted

Figure 13: 2D UNETR - attention rollout for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Rollout for the first (top), second (middle), and third (bottom) transformer blocks is plotted

Figure 15: 2D CATS - Compressed - only the first three transformer blocks from the original 2D CATS are retained. The convolutional backbone - a ResNet-50 in this figure - processes the image by gradually decreasing its spatial dimensions until they reach 14×14. This is added to a convolved output from the third transformer block and then goes through a series of upsamplings. The unsampled output is then passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. There is also an independent convolutional branch connecting the image directly to the final decoder layer. The numbers in the figure represent the channels from each stage of the process
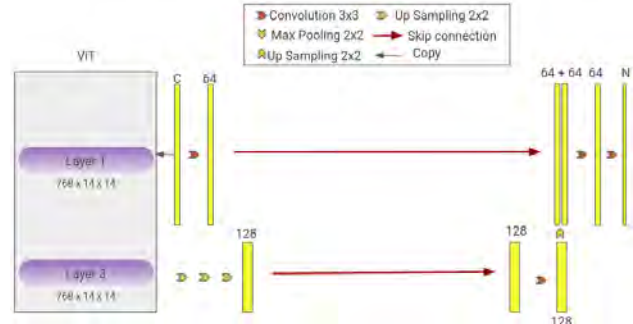
Figure 16: 2D UNETR - Compressed - only the first three transformer blocks from the original 2D UNETR are retained. The output from the third block is passed through a series of upsamplings. The upsampled output from the third block is passed on to the decoder which performs the process of upsampling and concatenating skip connections from the corresponding encoder output. There is also an independent branch connecting the image directly to the final decoder layer. The numbers in the figure represent the channels from each stage of the process

Figure 14: 2D CATS - attention rollout for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Rollout for the first (top), second (middle), and third (bottom) transformer blocks is plotted

indicate a global receptive field. While theoretically, a transformer model should be able to indicate a global receptive field even within the first block, this is not what is observed in the present case. Taking this information into account, compressed versions of both 2D CATS and 2D UNETR are proposed as can be seen in Figures. 15 and 16 respectively.

While transformer blocks from four to twelve can potentially provide more complex representations, they don't seem to be more informative in terms of receptive field. Hence, in both the proposed models, only the first three transformer blocks are retained and the rest discarded.

For the TransUNet, attention maps seem to indicate that spatial proximity plays no role even in the first and second transformer blocks. Hence, two compressed ver-
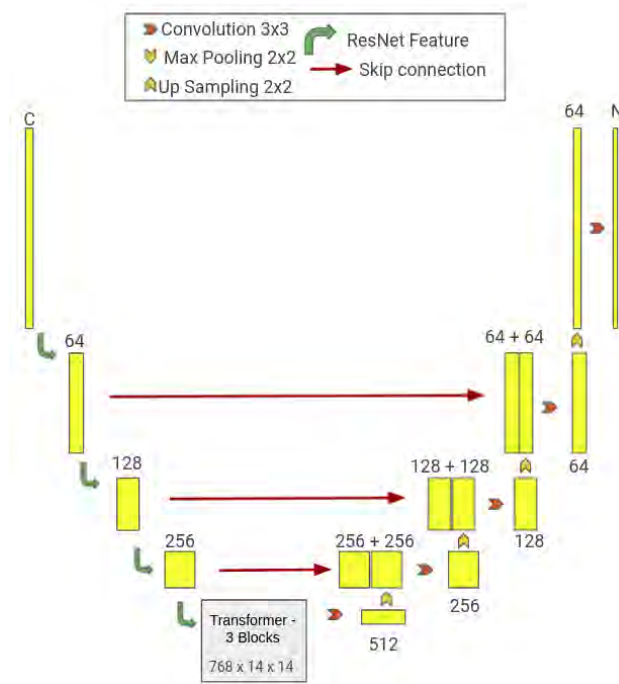
Figure 17: TransUNet - Compressed architecture - there is no change in the convolutional blocks in either the encoder or the decoder. The only change is in the number of transformer blocks. In one case, the number of retained transformer blocks is three, and in another case it is one. The numbers in the figure represent the channels from each stage of the process

sions are proposed. In the first version, instead of having twelve transformer blocks, the model has only three. In a second version the model is further compressed to have only one transformer block.

3.9.1. Attention Information in Compressed Models

Visualization of attention maps reveals that similar to the original TransUNet (Figure. 18), starting from the first transformer block, patches pay attention to other patches without being affected by the spatial proximity of those patches.

Similarly, the compressed 2D UNETR model (Figure. 19) closely follows the behaviour of the uncompressed one in that patches primarily only pay attention to themselves in the first block, extend attention to those patches within close proximity in the second block, and finally start paying attention to patches irrespective of spatial proximity in the third block.

For the compressed CATS model (Figure. 20), attention maps seem to follow the behaviour of the uncompressed model in the first block but not in the second. Whereas in the uncompressed model, patches pay attention to those within close proximity, in the compressed model, patches start paying attention without any specific regard for spatial proximity starting from the second block.
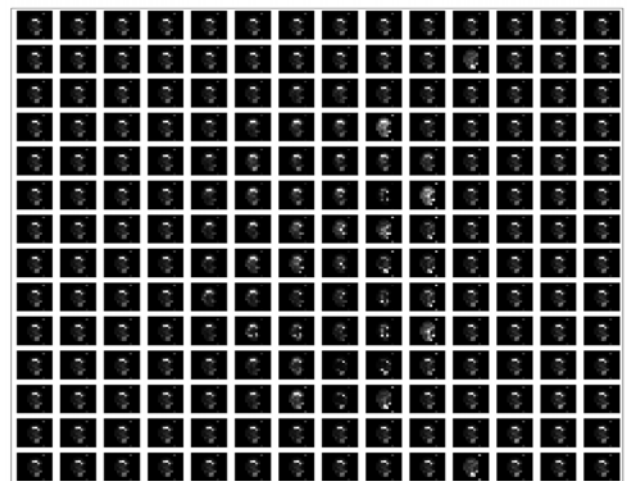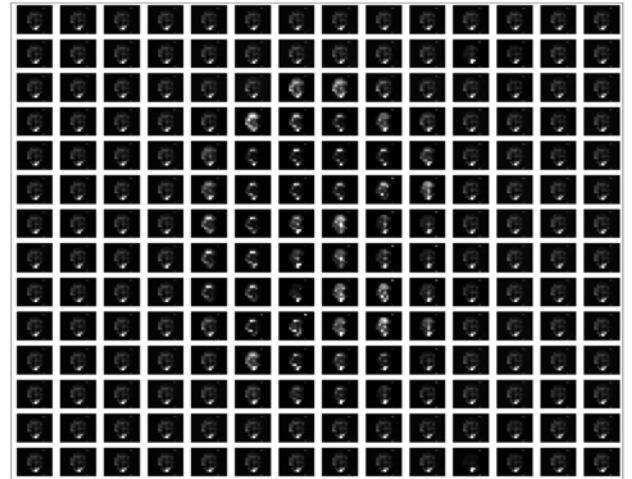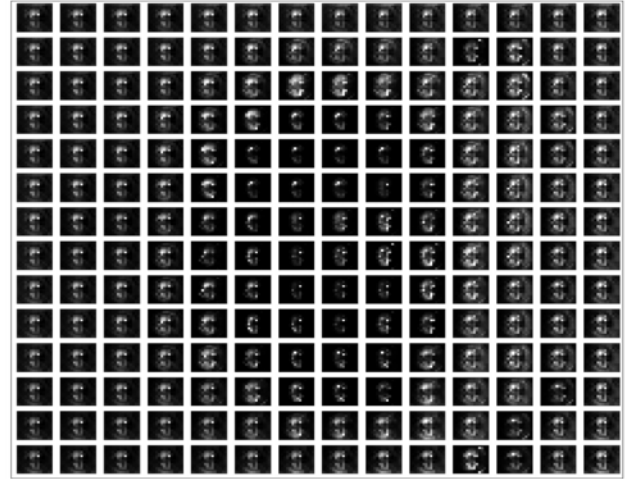
Figure 18: Compressed TransUNet - three transformer blocks - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Maps for the first (top), second (middle), and third (bottom) transformer blocks are plotted
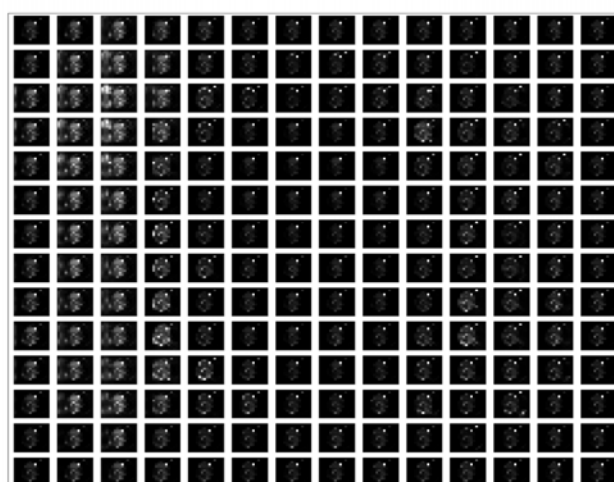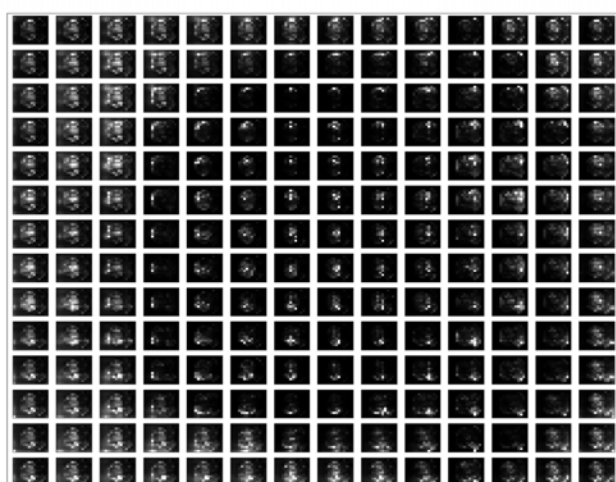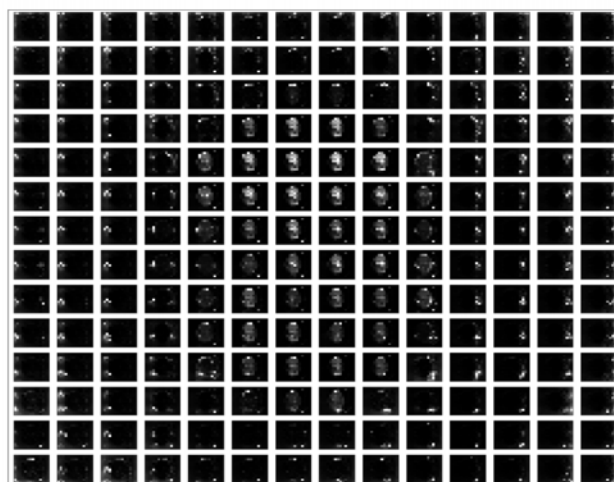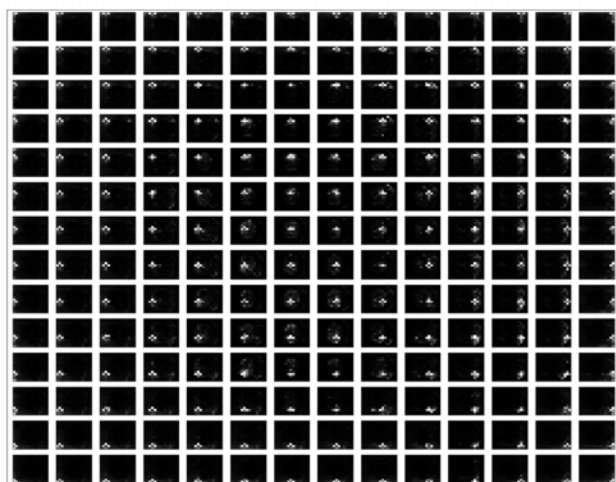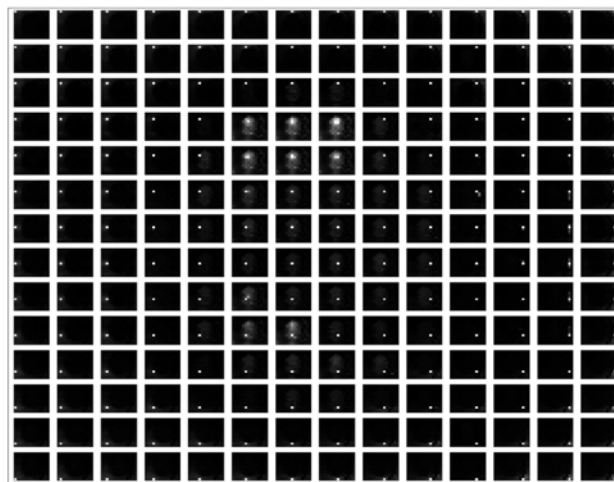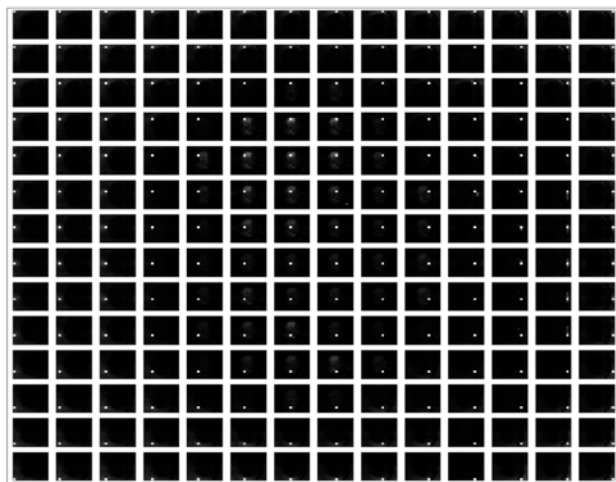
Figure 19: Compressed 2D UNETR - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Maps for the first (top), second (middle), and third (bottom) transformer blocks are plotted

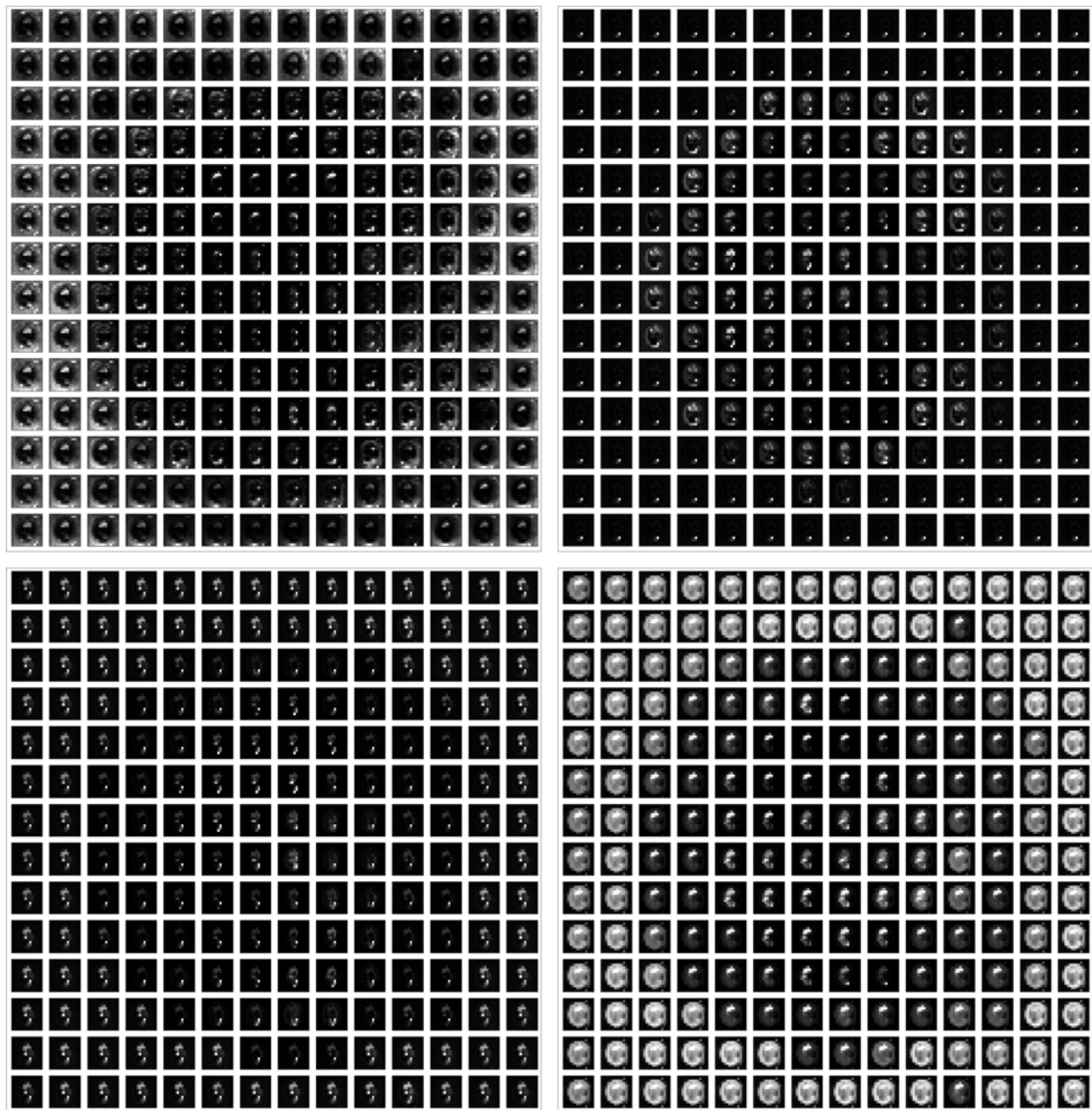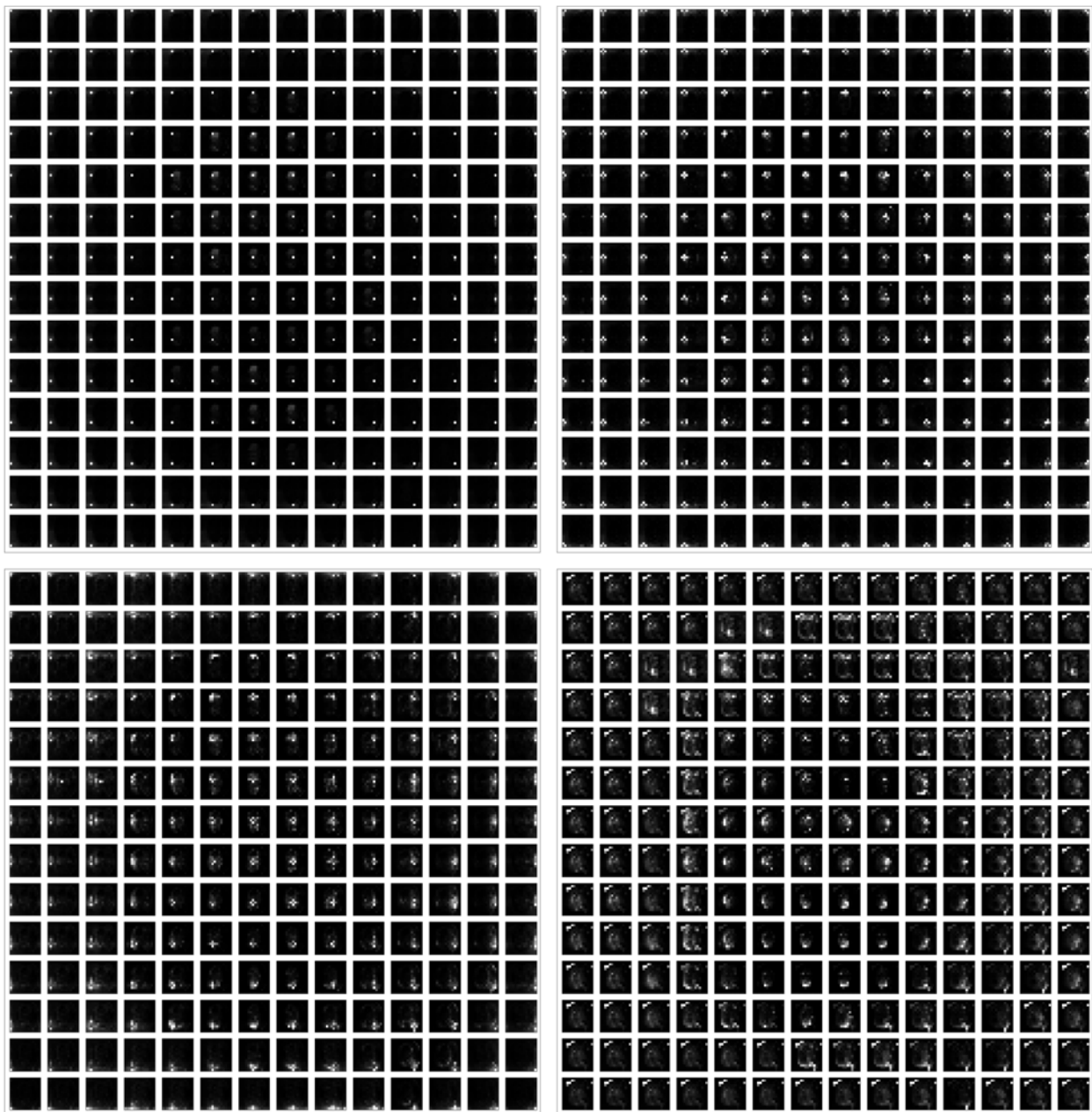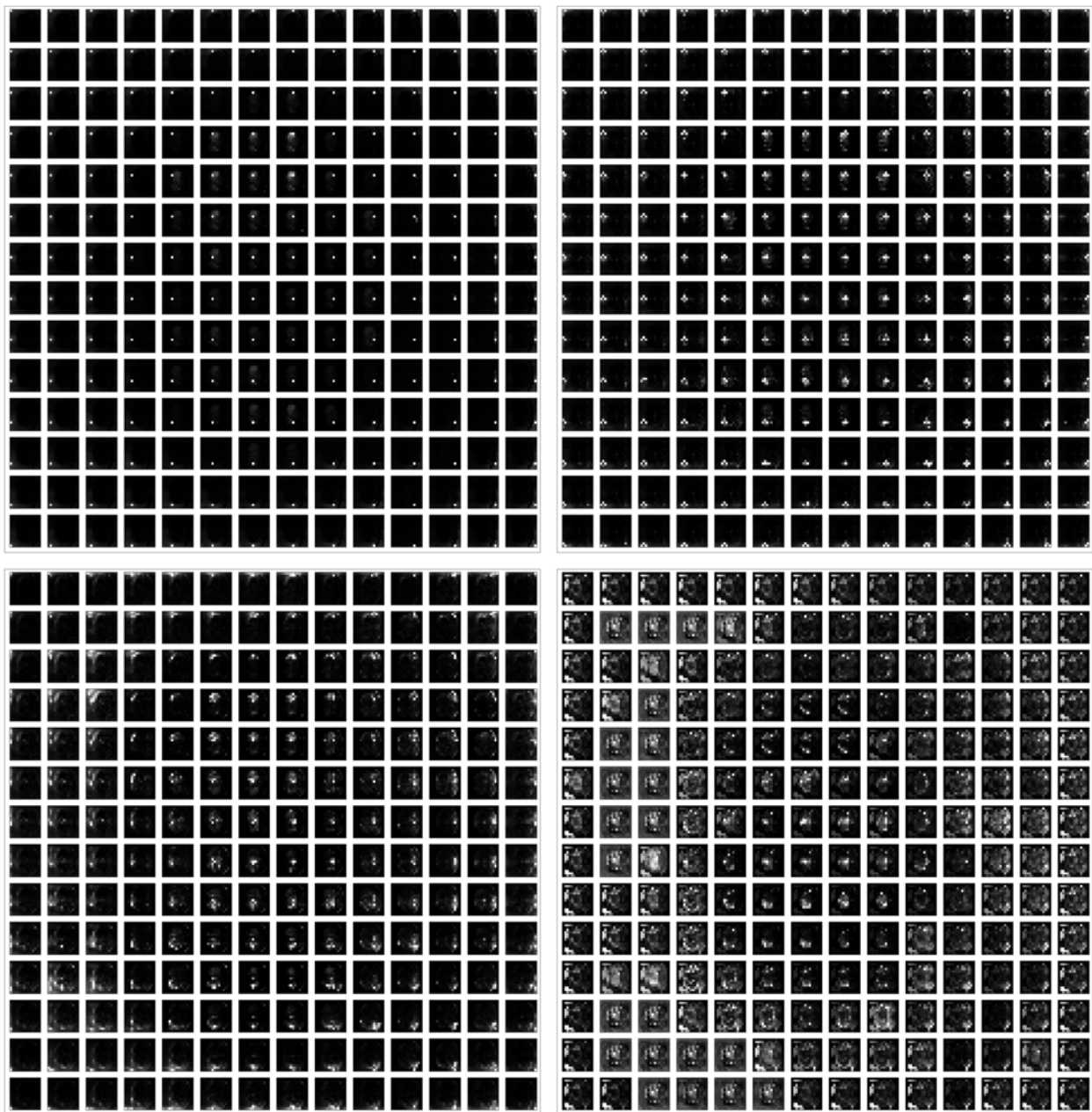Figure 20: Compressed 2D CATS - three transformer blocks - raw attention maps for a sample image from Synapse multi-organ dataset. Image reads from top to bottom. Maps for the first (top), second (middle), and third (bottom) transformer blocks are plotted

4. Results

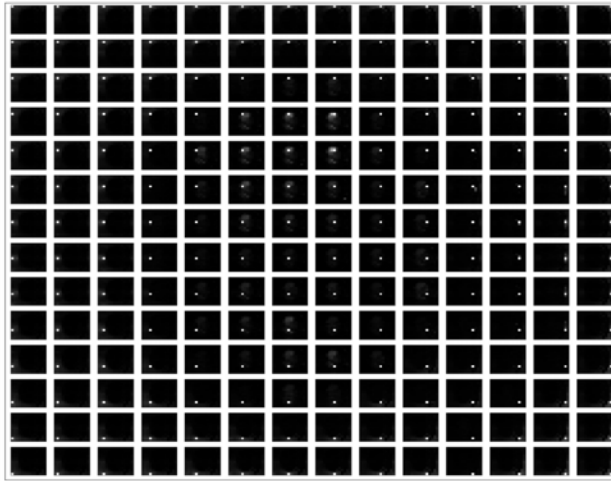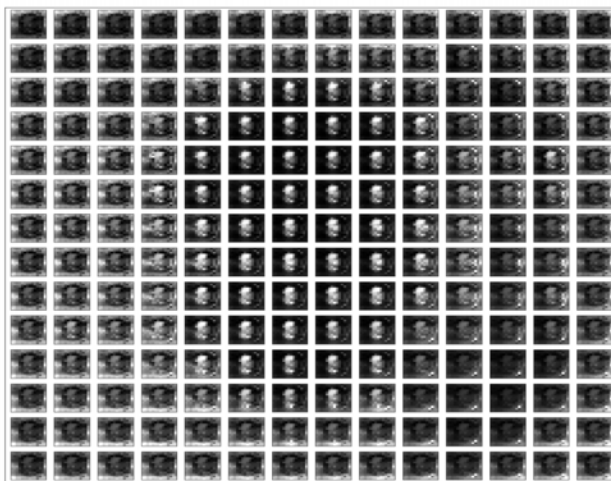The dice metrics for the uncompressed and the compressed models can be seen in Table. 1 for IBSR 18 dataset, Table. 2 for EMIDEC dataset, and Table. 3 for Synapse multi-organ dataset.

In addition to the dice metrics, the tables also indicate the percentage change in model performance going from an uncompressed model to a compressed one:

$$\% \ Change = 100 \times \frac{PC - PuC}{PuC} \qquad (6)$$

Where PC is the performance of the compressed model and PuC is the performance of the uncompressed model.

Lastly, the table also indicates the percentage saving in model parameters going from an uncompressed model to a compressed one:

$$\% \ Change = 100 \times \frac{\#uC - \#C}{\#uC} \qquad (7)$$

Where $\#C$ are the number of parameters in the compressed model and $\#uC$ are the number of parameters in the uncompressed model.

5. Discussion

Results seem to indicate it is indeed possible to compress segmentation models incorporating transformers without a drastic drop in performance. In addition to this, an analysis of information flow - particularly attention maps - can be utilized in order to steer this compression process.

The compressed versions of all three original models (TransUNet, 2D CATS, 2D UNETR) have less than 50% of the original parameters. The compressed version of 2D UNETR and the single block TransUNet have, in fact, less than one-third of the original parameters. In addition to this, it should also be noted that the choice of backbone architecture in the 2D CATS model can also influence the parameter saving process. The backbone utilized for the EMIDEC dataset was "ResNet-34", for IBSR 18 dataset it was "ResNet-50", and for the Synapse multi-organ dataset it was "DenseNet-121".

Model compression can be useful in multiple ways. It can help minimize computing resources while training. This, in turn, implies a reduction in energy consumption which has been highlighted as a major concern since the inception of the transformer model. Additionally, compressed models also stand a better chance when it comes to being deployed on mobile devices such as the Raspberry Pi which are generally low when it comes to storage. Lastly, whereas conventional transformer based models often limit the participation of the average AI researcher due to a lack of resources, model compression is a potentially useful step towards increasing

participation in transformer based research, essentially contributing towards the democratization of AI.

6. Conclusions

An analysis of raw attention maps and attention rollout revealed that for 2D CATS and 2D UNETR, in the first and second blocks, transformers behave similar to convolutional filters in that attention depends strongly on spatial proximity of patches. From the third block onward, this behavior changes and the transformer behavior can be characterized as having achieved a global receptive field. For the TransUNet, however, starting from the first block, attention behaves such that it is not limited by spatial proximity, and can be characterized as already having achieved a global receptive field.

Based on this analysis, model compression was performed such that all blocks after the first such block which achieves a global receptive field were discarded.

Based on the compressed models it can be argued that attention information from transformer blocks is helpful not only towards analyzing information flow, but it can also influence architectural decisions leading to model compression without seriously sacrificing model performance.

Model	Transformer Blocks	Dataset Size	CSF	Gray Matter	White Matter	Average Dice	% Dice Change (Compressed vs Uncompressed)	% Parameter Saving (Compressed vs Uncompressed)
TransUNet	12	1200	0.816	0.894	0.870	0.860	-	-
TransUNet - Compressed	3	1200	0.809	0.897	0.874	0.860	0	60.59
TransUNet - Compressed	1	1200	0.828	0.896	0.873	0.865	0.58	74.06
2D CATS	12	1200	0.820	0.893	0.873	0.862	-	-
2D CATS - Compressed	3	1200	0.812	0.902	0.877	0.864	0.19	61.15
2D UNETR	12	1200	0.815	0.901	0.883	0.866	-	-
2D UNETR - Compressed	3	1200	0.814	0.902	0.877	0.864	-0.21	75.12
2D UNETR - all trans	12	1200	0.809	0.900	0.875	0.861	-	-

Table 1: Dice metrics for IBSR 18 dataset. For the 2D CATS and 2D CATS - Compressed models, the backbone utilized was ResNet-50

Model	Transformer Blocks	Dataset Size	Myocardium	Infarction	NoReflow	Average Dice	% Dice Change (Compressed vs Uncompressed)	% Parameter Saving (Compressed vs Uncompressed)
TransUNet	12	558	0.846	0.654	0.774	0.758	-	-
TransUNet - Compressed	3	558	0.853	0.676	0.749	0.760	0.25	60.59
TransUNet - Compressed	1	558	0.859	0.680	0.767	0.769	1.46	74.05
2D CATS	12	558	0.855	0.569	0.674	0.699	-	-
2D CATS - Compressed	3	558	0.846	0.661	0.652	0.720	2.90	65.42
2D UNETR	12	558	0.841	0.593	0.723	0.719	-	-
2D UNETR - Compressed	3	558	0.830	0.617	0.663	0.703	-2.16	75.12
2D UNETR - all trans	12	558	0.846	0.574	0.743	0.721	-	-

Table 2: Dice metrics for EMIDEC dataset. For the 2D CATS and 2D CATS - Compressed models, the backbone utilized was ResNet-34

Model	Transformer Blocks	Dataset Size	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach	Average Dice	% Dice Change (Compressed vs Uncompressed)	% Parameter Saving (Compressed vs Uncompressed)
TransUNet	12	2212	0.873	0.662	0.869	0.826	0.944	0.612	0.912	0.843	0.818	-	-
TransUNet - Compressed	3	2212	0.878	0.667	0.874	0.842	0.948	0.665	0.891	0.826	0.824	0.73	60.59
TransUNet - Compressed	1	2212	0.881	0.643	0.859	0.831	0.947	0.618	0.916	0.836	0.816	-0.15	74.05
2D CATS	12	2212	0.848	0.664	0.857	0.818	0.935	0.626	0.893	0.838	0.810	-	-
2D CATS - Compressed	3	2212	0.862	0.671	0.855	0.803	0.939	0.567	0.879	0.732	0.788	-2.65	63.12
2D UNETR	12	2212	0.853	0.693	0.861	0.773	0.947	0.603	0.903	0.768	0.800	-	-
2D UNETR - Compressed	3	2212	0.853	0.656	0.832	0.760	0.950	0.568	0.888	0.783	0.786	-1.72	75.12
2D UNETR - all trans	12	2212	0.836	0.587	0.855	0.814	0.938	0.590	0.852	0.756	0.778	-	-

Table 3: Dice metrics for Synapse-multiorgan dataset. For the 2D CATS and 2D CATS - Compressed models, the backbone utilized was DenseNet-121

References

Chefer, H., Gur, S., Wolf, L., 2021. Transformer interpretability beyond attention visualization. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) , 782–791.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, L.A., Zhou, Y., 2021. Transunet: Transformers make strong encoders for medical image segmentation .

Desai, S., Ramaswamy, G.H., 2020. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. 2020 IEEE Winter Conference on Applications of Computer Vision (WACV) , 972–980doi:10.1109/WACV45572.2020.9093360.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale. ICLR .

Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H., Xu, D., 2022a. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images .

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, A.B., Roth, R.H., Xu, D., 2022b. Unetr - transformers for 3d medical image segmentation. WACV , 1748–1758.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 770–778.

Huang, G., Liu, Z., Maaten, v.d.L., Weinberger, Q.K., 2017. Densely connected convolutional networks. 30TH IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017) , 2261–2269.

Krizhevsky, A., Sutskever, I., Hinton, E.G., 2017. Imagenet classification with deep convolutional neural networks. Commun. ACM , 84–90.

Lalande, A., Chen, Z., Decourselle, T., Qayyum, A., Pommier, T., Lorgis, L., Rosa, d.l.E., Cochet, A., Cottin, Y., Ginhac, D., Salomon, M., Couturier, R., Meriaudeau, F., 2020. Emidec: A database usable for the automatic evaluation of myocardial infarction from delayed-enhancement cardiac mri. international conference on data technologies and applications doi:10.3390/data5040089.

Li, C., Tan, Y., Chen, W., Luo, X., Gao, Y., Jia, X., Wang, Z., 2020. Attention unet++: A nested attention-aware u-net for liver ct image segmentation. 2020 IEEE International Conference on Image Processing (ICIP) , 345–349doi:10.1109/ICIP40778.2020.9190761.

Li, H., Hu, D., Liu, H., Wang, J., Oguz, I., 2022. Cats: Complementary cnn and transformer encoders for segmentation, in: 2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI), pp. 1–5. doi:10.1109/ISBI52829.2022.9761596.

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization. ICLR .

Milletari, F., Navab, N., Ahmadi, S.A., 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. Proceedings of 2016 Fourth International Conference on 3D Vision (3DV) , 565–571doi:10.1109/3DV.2016.79.

Oktay, O., Schlemper, J., Folgoc, L.L., Lee, C.H.M., Heinrich, P.M., Misawa, K., Mori, K., McDonagh, G.S., Hammerla, Y.N., Kainz, B., Glocker, B., Rueckert, D., 2018. Attention u-net: Learning where to look for the pancreas. arXiv: Computer Vision and Pattern Recognition .

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. Lecture Notes in Computer Science , 234–241.

Samira, A., Willem, Z., 2020. Quantifying attention flow in transformers. ACL , 4190–4197doi:10.18653/v1/2020.acl-main.385.

Schlemper, J., Oktay, O., Schaap, M., Heinrich, P.M., Kainz, B., Glocker, B., Rueckert, D., 2019. Attention gated networks: Learning to leverage salient regions in medical images. Medical Image Analysis , 197–207.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2020. Grad-cam: Visual explanations from deep networks via gradient-based localization. International Journal of Computer Vision , 336–359doi:10.1109/ICCV.2017.74.

Tan, M., Le, V.Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. International Conference on Machine Learning , 6105–6114.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, N.A., Kaiser, L., Polosukhin, I., 2017. Attention is all you need. Advances in Neural Information Processing Systems 30 (NIPS 2017) , 5998–6008.

Wang, W., Chen, C., Ding, M., Yu, H., Zha, S., Li, J., 2021. Transbts: Multimodal brain tumor segmentation using transformer. Medical Image Computing and Computer Assisted Intervention - MICCAI 2021, PT I , 109–119doi:10.1007/978-3-030-87193-2_11.

Wightman, R., 2019. Pytorch image models. https://github.com/rwightman/pytorch-image-models. doi:10.5281/zenodo.4414861.

Xie, Y., Zhang, J., Shen, C., Xia, Y., 2021. Cotr: Efficiently bridging cnn and transformer for 3d medical image segmentation. Medical Image Computing and Computer Assisted Intervention - MICCAI 2021, PT III , 171–180doi:10.1007/978-3-030-87199-4_16.

Xu, G., Wu, X., Zhang, X., He, X., 2021. Levit-unet: Make faster encoders with transformer for medical image segmentation .

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A., 2016. Learning deep features for discriminative localization. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 2921–2929.

Zhou, Z., Siddiquee, M.R.M., Tajbakhsh, N., Liang, J., 2018. Unet plus plus : A nested u-net architecture for medical image segmentation. Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, DLMIA 2018 , 3–11.

Learning Cytoarchitectonic Structure From 3D Polarized Light Imaging

S M Ragib Shahriar Islam [1,2], Alexander Oberstraß[1], Esteban Vaca[1], Markus Axer[1], Timo Dickscheid[1]

[1]*Institute of Neuroscience and Medicine, Forschungszentrum Jülich, Germany*

[2]*Escola Politecnica Superior, Universitat de Girona, Spain*

Abstract

Light microscopy of post-mortem brain sections is a prime image acquisition procedure for microscale brain tissue analysis. Over the last decade, 3D Polarized Light Imaging (3D-PLI) has been one of the most successful procedures for revealing the organization of nerve fibres inside brain microscopy samples (Axer et al., 2011). Thanks to the birefringence property presented in the myelin sheaths surrounding the axons, the polarization microscope can measure the nerve fiber orientation while a polarized light beam passes through the brain sections. Such a procedure reveals the orientation description of the fiber tracts at the sub-millimeter level (mesoscale). One advantage of 3D-PLI acquisition is that it does not require tissue staining. Therefore, it is ideally suited for multi-modal analysis by combination, e.g., with staining for cell bodies after the 3D-PLI measurement. Nevertheless, the acquisition process of such multi-modal data is challenging and time-consuming, enabling access to only a limited number of samples. This study aimed to investigate which cytoarchitectonic features are already inherent in the 3D PLI data, circumventing the costly data acquisition, using scalable unsupervised deep learning methods like Variational Autoencoders (VAE) and Conditional Generative Adversarial Networks (GAN). This research method uses the parameter maps of 3D PLI images to forecast the light microscopic cytoarchitectonic image via a progressive process from Variational Autoencoders to region mutual information-based conditional GAN. The results of this thesis, both qualitative and quantitative, show that the proposed method has a lot of potential for predicting Cytoarchitectonic images from 3D-PLI images. Between the real and generated cytoarchitectonic images, the Mean Squared Error, Universal Quality Index, Average Log-Likelihood, and Maximum Mean Discrepancy(with RBF kernel) exhibit correlated results.

Keywords: 3D Polarized Light Imaging, Cytoarchitectonic Structure, Fiber Tract Orientation, Variational Autoencoder, Deep-Learning, Region Mutual Information, Conditional GAN.

1. Introduction

1.1. Image Modality Transfer

Image to image transfer, also known as image modality transfer, is a relatively new study area in medical image analysis with a wide range of potential applications. In general, image to image transfer can be used for standard modality translation (e.g., PET to CT), motion correction, denoising(Armanious et al., 2020), (Zhou et al., 2020), radiation reduction, artifact correction(Vey et al., 2019), better image acquisition/estimation(Wang et al., 2018), data augmentation(Sorin et al., 2020) among other applications.

The purpose of this study is to use 3D-PLI to predict the cytoarchitecture of brain sections. The main goal is to find if image modality transfer can predict reasonable cell distributions from 3D-PLI, which is used to identify fiber orientation in brain sections.

1.2. Motivation

One of the most significant problems with microscopic images is the deformation of cells and tissues during sectioning and histological processing. The ultrastructural features of cells are extremely difficult to retain, and there is currently no method for doing so. This phenomenon causes some inaccuracy in the analysis of brain cell structure and during the creation of brain atlases.(Axer et al. (2011), Wilson and Bacic

(2012)).

Performing microscopic image acquisition of two different modalities from the same section could involve many problems with the samples, given the amount of manipulation that is involved in handling the tissue from one modality to another. Staining microscopy is a tedious and lengthy process. After the image acquisition by 3D-PLI microscopy unmounting from the PLI imaging stage, cleaning, staining, re-measured, and then again registering to the PLI is a long-time consuming process. Such manipulation, could inevitably damage the tissue. Therefore, having an alternative way to predict a modality from another constitutes is a time, energy, and money-saving approach.

These facts were the driving force for this study, which used PLI images to construct Cytoarchitectonic images. The primary goal is to determine whether the PLI images had enough information to predict Cytoarchitectonic images using the generative models.

As 3D-PLI performs multiple image acquisitions at different polarization angles, we investigated if such data contains information about the cell density present in brain samples(Axer et al., 2011). To accomplish such a task, we leverage the recent advances in image generation performed by Conditional Generative Adversarial Networks (cGANs).

It could be useful to make a synthetic backup copy of the PLI dataset's Cytoarchitectonic images. Moreover, it will be useful in case tissue is damaged lost or cannot be stained.

1.3. Microscopy Modalities

This work uses two modalities of post-mortem vervet monkey brain section images. The 3D-Polarized Light Images acquired as described in Axer et al. (2011) at Forschungszentrum Jülich, and another modality is stained light microscopic histology images, also known as Cytoarchitectonic images. Both the modality images were acquired from the same vervet monkey brain and of the same corresponding sections.

1.3.1. Polarized Light Image

Brain's structure and function are deeply entwined at different levels of brain tissue interconnection. Obtaining a precise image of postmortem brain sections to investigate their 3D fiber structure and fiber tract orientation is thus challenging. Axer et al. (2011) developed a novel image acquisition technology for postmortem human brain sections utilizing 3D polarized light as a result of the perception to address these issues. When a polarized light beam passes through the myelin sheath around the axons, this approach utilizes the birefringence property of the myelin sheath, see Figure 1.
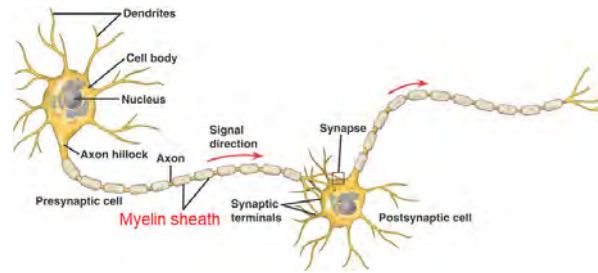
Figure 1: Basic neuron parts. From left to right: dendrites, which collect other neuron impulses; the cell body, where the nucleus is allocated; the axon, the channel that sends the information to the postsynaptic cell; and finally the synapse, the union between the two cells needed for the communication(Grau-Moya (2011)). The birefringence property shown by the cylindrical coating of Myelin sheaths(red font in the image) around the Axons are the main basis of Axer et al. (2011)'s 3D-PLI imaging technique.

The regular distribution of lipids and proteins in the myelin sheath causes birefringence in nerve fibers, which results in specific optical anisotropy. The net birefringence of the neurofilaments inside the axon and the radially oriented lipid chains of the myelin sheath can be explained by a single axis of optical anisotropy, which produces uniaxial negative birefringence and hence reflects the fiber's spatial orientation (de Campos Vidal et al., 1980).

Image Acquisition Setup:
The physical configuration for the 3D PLI image acquisition method as described in Axer et al. (2011) is shown in Figure 2. A monochromatic green LED light source was employed as light source. After the LED, a first polarizer is placed, which converts the incoherent light into polarized light. The imaging brain section was then held on a specimen stage. Then, a retarder and another polarizer for experimenting with birefringence characteristics and setting the reference at various polarizing angles.

Image Acquisition Methodology:
The configuration of Polarizing Microscopy in Axer et al. (2011) is explicitly designed to obtain the image for various retarder or polarizer rotation angles. The rotation angle, ρ, was adjusted by rotating the retarder in $10°$ increments from $0°$ to $170°$. As a result, there were a total of 18 photos for 18 various rotation angles.

This study used the brain of a vervet monkey, labeled as monkey-1818 (male, 2.4 years old) described in Takemura et al. (2020). A polarimetric microscopy arrangement based on a Köhler lit (wavelength spectrum: 550 nm± 5nm) bright field microscope fitted with two polarizing filters and a moving specimen stage as described in Reckfort et al. (2015) was used to perform microscopic imaging referred to as 3D-PLI in Axer et al. (2011). The monochromatic CCD camera had a field of view of 2.7 x 2.7 mm^2 and a 1.3μm in plane pixel resolution.
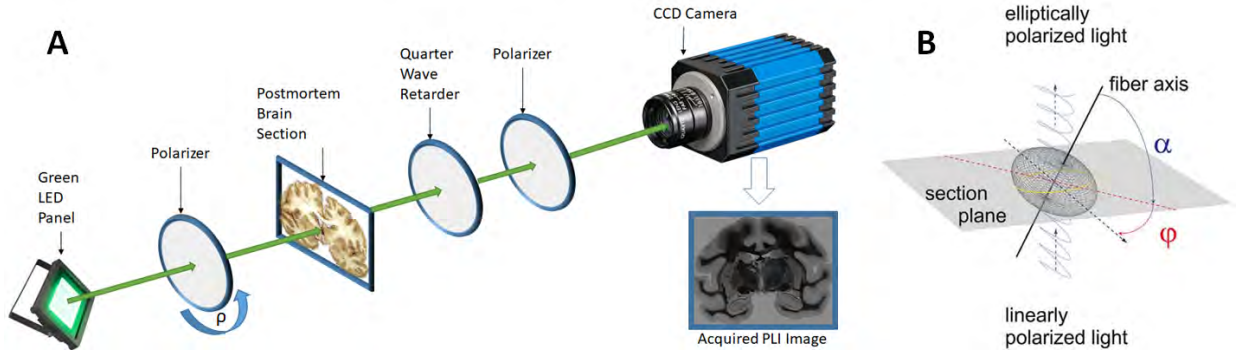
Figure 2: (A) Optical stup diagram of PLI image acquisition process. (B) The optical fiber model is depicted in this diagram. An elliptically shaped oblate surface, the refractive index ellipsoid or indicatrix, describes the refractive index of a negative uniaxially birefringent medium, such as a myelinated axon (gray mesh). A single axon's myelin sheath (black line) reacts locally with a beam of linearly polarized light (blue trace), causing the light beam to phase shift. The light becomes elliptically polarized, and it can be used to determine the indicatrix's orientation or the predominant local fiber orientation. The in-plane direction angle ϕ and the out-of-section inclination angle α in the frame coordinate system determine this orientation.(Axer et al., 2011)

Therefore, imaging entire brain sections with vast areas needed tile-wise scanning with 1mm (coronal series) overlaps. During the measurement, unstained sections were exposed to linearly polarized light, and the intensity of the transmitted light was collected using a circular analyzer(Takemura et al., 2020). However, in this study we have used 9(out of 18) of these rotation angle images at $20°$ apart to save memory during the training process.

Therefore, according to Reckfort et al. (2015) the intensity profile $I_T(\rho)$ of a PLI measurement, given I_{T0} is the transmitted intensity of light after passing through the tissue and ρ different polarization angles can be described as:

$$\mathcal{I}_T(\rho) = \frac{\mathcal{I}_{T0}}{2}[1 + \sin(2\rho - 2\phi)r], \qquad (1)$$

where, with Section Thickness(t), Myelin birefringence(Δn), and light wavelength(λ) the r represents by,

$$r = \left| \sin\left(2\pi \frac{t\Delta n}{\lambda} \cos^2 \alpha\right) \right| \qquad (2)$$

After the 3D-PLI image acquisition, the following parameter maps were obtained by performing signal analysis:

The Transmittance map: This is a measure of light attenuation after passing through the polarimeter and brain tissue, and represents the pixel-wise average map of all PLI raw images.

The Retardation map: It describes the extent of the phase shift induced to the light wave due to interaction with the birefringent by approximating the normalized amplitudes of the light intensity profiles.

The Direction map: It specifies the in-section direction angle of each fiber, i.e. the x–y orientation.

The fiber inclination map: It refers to the vertical component of each fiber's out-of-section angle.

The map images can be calculated with discrete harmonic Fourier analysis(Glazer et al., 1996); (Axer et al., 2011).

Hence, We can parameterize equation (1), and we get,

$$\mathcal{I}_T(\rho) = a_0 + a_1 \sin(2\rho) + b_1 \cos(2\rho) \qquad (3)$$

with,

$$a_0 = \frac{\mathcal{I}_{T0}}{2}; a_1 = \frac{\mathcal{I}_{T0}}{2} r \cos(2\phi); b_1 = -\frac{\mathcal{I}_{T0}}{2} r \sin(2\phi). \qquad (4)$$

These coefficients(a_0, a_1, b_1) are computed for each pixel of the image from the measured intensity set $I_T(\rho_i)$. Thus, for N = 9 samples data points from 9 rotation angles(ρ) we get,

$$a_0 = \frac{1}{N}\sum_{i=1}^{N} \mathcal{I}_T(\rho_i); a_1 = \frac{2}{N}\sum_{i=1}^{N} \mathcal{I}_T(\rho_i) \sin(2\rho_i); \qquad (5)$$

$$b_1 = \frac{2}{N}\sum_{i=1}^{N} \mathcal{I}_T(\rho_i) \cos(2\rho_i)$$

Therefore, for each pixel of the image, we can retrieve the light retardation, light transmittance, and the quantified fiber orientation(α, ϕ) by combining these aforementioned Fourier coefficients(Reckfort et al. (2015)). Hence, we get the following three parameter maps,
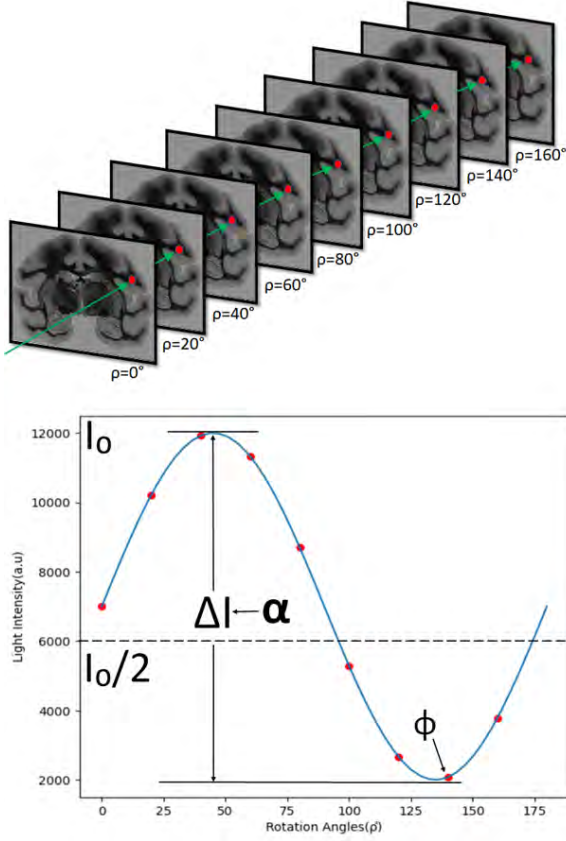
Figure 3: According to Axer et al. (2011), a typical PLI raw image data collection consists of 18 images with equidistant rotation angles ranging from 0° to 170°. In this study a selection of 9 coronal section images are used(images for each 20° increment). Given here, with the sketched arrow indicating one example pixel per image. The measured light intensities are examined pixel-by-pixel as a function of discrete rotation angles ρ to derive the fiber orientation. The developed physical model ties the sine phase to the direction angle *phi* and the amplitude to the inclination angle *alpha*, and offers a precise mathematical summary of the data (continuous blue line). The data points that have been highlighted in red correspond to the images that have been chosen.

$$Transmittance, \mathcal{I}_{T0} = 2a_0 \qquad (6)$$

$$Retardation, r = \frac{(a_1^2 + b_1^2)^{\frac{1}{2}}}{a_0} \qquad (7)$$

$$Direction, \phi = \frac{1}{2}\arctan 2(b_1, -a_1) + \frac{\pi}{2}$$
$$= \frac{1}{2}\arg(a_1 + ib_1) \qquad (8)$$

As we have three different parameterized maps for each section, it motivated our research work to investigate further to find if 3D PLI images preserve the information necessary to predict the cytoarchitectonic image for each corresponding section of the brain. The parameter map images are shown in Figure 4

1.3.2. Cytoarchitectonic Image

The spatial structure of neuronal cells in the brain, including their arrangement into layers and columns with respect to cell density, orientation, and the presence of specific cell types, is referred to as cytoarchitecture. It allows for the segmentation of the brain into cortical and subcortical nuclei, as well as the linking of structure, connection, and functions (Schiffer et al. (2021)).

Myeloarchitecture is the term for the organization of nerve fibers. The cytoarchitectonic and the myeloarchitecture as well organization of the brain is separated into six sections, as depicted in Figure 5

Cytoarchitecture Acquisition: After flushing the brain with phosphate buffered saline and perfusion fixing with 4 percent paraformaldehyde. The brain tissue was then soaked in 20% glycerin, deep frozen, and preserved at -70° Celsius. Monkey 1818's brain was sectioned coronally. A large-scale cryostat microtome with a 60 μm thickness was used to do serial sectioning(Takemura et al., 2020).Then the 3D-PLI imaging was performed. After the 3D-PLI image acquisition, cresyl violet Nissl staining was performed on brain sections. The stained section was then examined under a camera lucida microscope, and the boundaries between the cellular layers were delineated according to Braak (1980).(Zeineh et al., 2017) The acquired cytoarchitectonic image is shown in Figure 6.

2. State of the art

2.1. Image Modality Transfer

In a broad sense, the imaging modality refers to a process by which an image was acquired, as well as the various appearances of images obtained through different imaging techniques, such as images captured with a digital camera, a celluloid camera, a heat camera, an infrared camera, satellite footage, or even computer-generated or hand-drawn illustrations. In the medical imaging domain, here on the other side, imaging modalities are frequently classified by the physical principle by which images are generated: ultrasound, radiation such as x-rays, MRI among others. Every form of image has its own common features. Each imaging modality has its common features, which defines different properties such as, colour spectrum, contrast, shape or resolution(Jacques and Christe, 2020).

Therefore, image modality transfer refers to the process of creating an image from one modality to another. Creating a map from an aerial view, converting a zebra image to a horse, or generating a Computed Tomography (CT) image from a Positron Emission Tomography (PET) image are examples of image modality transfer. For computer vision engineers, this is a critical tool for saving money and time. It even assists doctors with prognosis or follow-up tests in the medical imaging sector, as well as researchers studying animal or plant
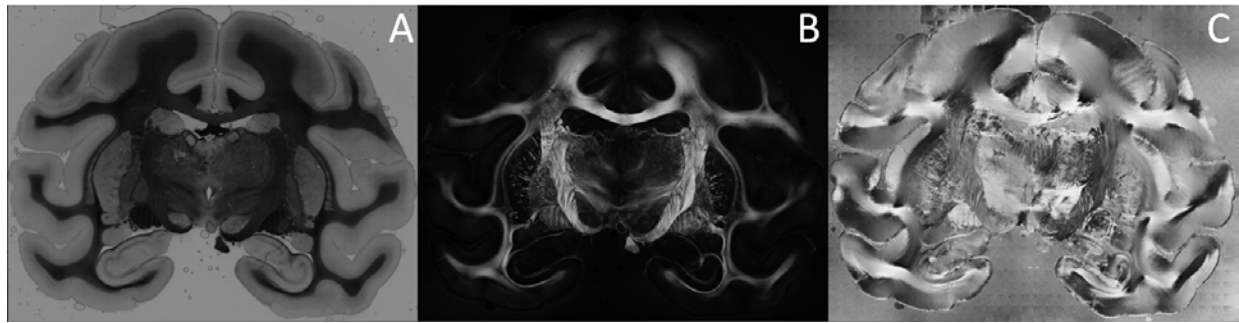
Figure 4: Representation of (A)Transmittance map, (B)Retardation map, and (C)Direction map of 3D-PLI image of a Vervet-monkey's brain coronal section-539.

tissues in new research dimensions.((Armanious et al., 2020)

2.1.1. On Non-medical Image Dataset

In the computer vision field, image modality transfer is currently a hot topic. A considerable amount of work on Generative Adversarial Networks(GAN) is being conducted for modality transfer. At the outset of this research, we considered all of the significant past publications relevant to our work to form a common understanding.

The mapping of every image object's key features, as well as the ambiguity of the mapping, are both hidden in a low-dimensional latent vector of any image modality. Zhu et al. (2017) have demonstrated how the ambiguity of the mapping can be randomly altered and sampled in a variety of ways, resulting in particular changes in the feature map changing the modality of the input image. A generator effectively learns to map the supplied input image to the output using the latent coding. The invertibility of the output image to the original input image must be preserved for the model to prevent many-to-one prediction. They used Conditional Variational Autoencoder GAN(cVAE-GAN), Conditional Latent Regressor GAN(cLR-GAN), and BicycleGAN to demonstrate how one-to-many modality transfer can be modulated. They've also displayed a comparison of their results. However, They were unable to fully control the output modality aspects. They discovered the same object's latent space in another modality, but they didn't disclose how to control the output in any of the latent spaces; instead, all of the outputs were generated at random.
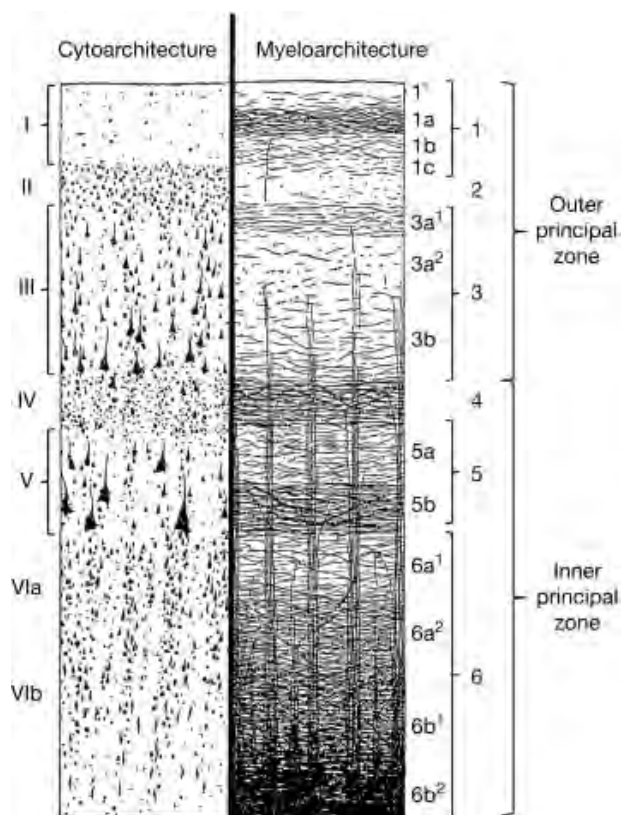
Figure 5: A Comparative illustration of different layers in cytoarchitectonic and myeloarchitectonic structure of brain. The Roman number indicate the layers in Cytoarchitecture and the Arabic numbers represent in Myeloarchitecture.(Zilles et al., 2015)
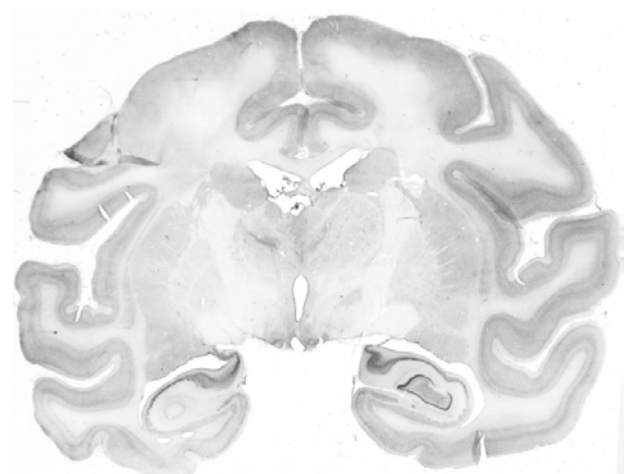
Figure 6: The stained microscopic image of the Vervet-monkey's brain section-539.

A shared latent distribution exists between images of the same object from two different modality (for example, a picture of a cat from two different breeds). A given set of (Cat_1, Cat_2) from a joint distribution $P_{Cat_1, Cat_2}(Cat_1, Cat_2)$ are used in supervised learning. Liu et al. (2017) have found the marginal shared distributions in the Gaussian latent space using two different GANs that were entirely unsupervised (So the resultant marginal distribution is $P_{Cat_{General}} = P_{Cat_1}(Cat_1) \cap P_{Cat_2}(cat_2)$).

To shift the modality of the images, Liu et al. (2017) used a Gaussian latent space assumption on this generalized distribution. They were, however, constrained by two major factors. The training was unstable due to the saddle point finding challenge, and the translation was unimodal due to the assumption. As a result, the generated images include a lot of artefacts.

According to the requirements of our research, we must predict an image of a specific modality (Cytoarchitecture). Therefore, we must set up certain conditional parameters. In Isola et al. (2017), as seen in Figure 7, they used a condition x to generate an image y from a latent noise vector z.

Equation (9) is Isola et al. (2017) model's loss function. Although the Generator(G) seeks to minimize the objective versus an adversary Discriminator(D) that wants to maximize the objective, the conditional vector x directs both the G and D to train on a specific condition.

$$\mathcal{L}_{cGAN}(G, D) = E_{x,y}[\log D(x, y)]$$
$$+ E_{x,z}[\log(1 - D(x, G(x, z)))] \qquad (9)$$

Isola et al. (2017) also have used the L1 distance with the adversarial loss as L1 encourages less blurring effect. This way the discriminator's job stays identical, but the generator must not only deceive the discriminator, but also be get close to the ground truth output in an L1 distance sense.

$$\mathcal{L}_{L1}(G) = E_{x,y,z}[\|y - G(x, z)\|_1] \qquad (10)$$

Therefore, the final goal of the model can be expressed as following.
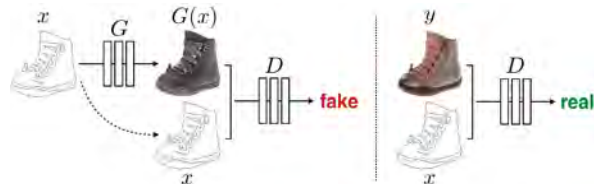
Figure 7: Training a conditional GAN to map edges→photo. The discriminator, D, learns to classify between fake (synthesized by the generator) and real edge, photo tuples. The generator, G, learns to fool the discriminator. Unlike an unconditional GAN, both the generator and discriminator observe the input edge map.(Isola et al. (2017))

$$G = arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \qquad (11)$$

We adopted much of the concepts from Isola et al. (2017) paper in our own research, though they used noise as an input, which we didn't use. Rather we have used the style transfer approach.

2.1.2. On Medical Image Dataset

As previously stated, current research on modality transfer in medical image analysis is context-specific. Image translation techniques have mostly been used to transfer an image from one modality to another, as well as for denoising, motion correction, and other purposes. The lack of sufficient data, as well as the high accuracy and efficiency (callback) necessary in medical domain research, are the bottlenecks in this study field(Xie et al. (2021)). Additionally, brain image analysis at the cell level necessitates a significant amount of computational power, which continues to be a barrier in this field of this research(Shen et al. (2017)).

Xiang et al. (2018) attempted to create synthetic CT scan images from T1-weighted MRI scans for both brain and prostate imaging. They made use of the Deep Embedded Convolutional Neural Network (DECNN) idea. They created feature maps from MRI scans first and then converted these feature maps using Deep CNN. They claimed that their model outperformed the state-of-the-art models mentioned in their research. However, their research was based on the images' higher-level features such as contrast, edges, shapes. In contrast, our work is more focused on the lower level detailed features as well as the higher level feature maps, as we have to predict the cytoarchitecture from 3D-PLI images, preserving the regional information of the brain tissues from the different brain parts.

In Wei et al. (2019), they used multimodal MRI imaging such as Magtizing Transfer Imaging (MTI MRI), Diffusion Tensor Imaging of T1-weighted, Radial Dissusivity, and Fractional Anistropy (DTI-T1, DTI-RD, DTI-FA) to predict PET-demyelination. They used 3D-U-net architecture to construct a feature map from different four modalities, and then used an adversarial network to predict the PET. However, using 9/10/11 - D picture inputs, we used their concept of employing a U-net-based auto-encoder. Although their findings are promising, they are incompatible with cell-level accuracy. They were effective in obtaining higher-level PET images.

Armanious et al. (2020) detailed how they combined non-adversarial loss with the adversarial loss for image prediction in another modality in the MedGAN article. They've launched a new Generator architecture known as CasNET, which is fundamentally a Resnet-based U-net architecture with skip connections. They planned for the Generator to be a general-purpose application

rather than a task-specific one. However, this approach limits the purpose of our study because it loses a lot of the image's detailed description during modality transfer. They used the Discriminator network as a trainable feature extractor, which was used to train both the Generator and the Discriminator using perceptual loss. They demonstrated the use of their network in PET to CT modality transfer, MRI motion noise correction, and PET image denoising. However, in our circumstance, we require an application-specific model with a high cell-level detail accuracy.

3. Material and methods

3.1. Description of the Dataset

The 3D-PLI image (taken using the approach described in Axer et al. (2011)) and related cytoarchitectonic images of Vervet monkeys were used in this study. In terms of memory and size, the 3D-PLI image and cytoarchitectonic image sets that we analyzed in our research are massive. As a consequence, we generated a small segment of the main dataset for the part of developing a functional deep learning model for the experiment. Once a solid deep learning model was developed, we proceeded to train the models with the main large dataset to improve the model's accuracy. Using the main dataset, for example, to only discover the optimum operational deep learning model will be computationally exorbitant (even for the supercomputing facility available in Forschungszentrum Jülich).

3.1.1. Original Dataset

The 3D-PLI image and Cytoarchitectonic image pairs of each brain section available at Forschungszentrum Jülich are multidimensional Big data. Hence, they are stored in Hierarchical Data Format version 5 (HDF5) format.

Polarized Light Image:
The dimension for 3D-PLI images varies for different sections. In this study each 3D-PLI image for a single section has the dimension of (17715, 22865, 9) which is a downsampled version of the original 3D-PLI by factor 2 at resolution of 2.6 μm. where it represents(#rows of pixels, #columns of pixels, #images for different rotation angles) and each image size in terms of memory is around 3 GigaBytes(GBs). The original dataset contains 9 3D-PLI, Cytoarchitectonic image pairs. Hence, the Total raw dataset size becomes around 3×9 = 27GBs for 3D-PLI images.

Cytoarchitectonic Images:
The original in pair Cytoarchitectonic images for the 3D-PLI images were obtained a higher resolution 1μm than the PLI images. Each corresponding Cytoarchitectonic image is downsampled such that they also have the same dimension of (17715, 22865) as (#rows of pixels, #columns of pixels) occuping about 2 GBs in memory size. Therefore, total size for the cytoarchitectonic dataset is about $9 \times 2 = 18$GBs.

Additional Parameterized Map Images and Masks :
Moreover, After the preprocessing, we created the mask images for each PLI image to train the model without the background of the images which is also of the same dimension as cytoarchitectonic images(17715, 22865) and each mask takes a memory size of about 400MegaBytes(MB). The Transmittance, Direction and Retardation maps of the same dimension of approx. 3GB each.

Therefore, we have additional training data of Total 400MB $\times$ 9 = 3.6 GB for the mask images, 3 $\times$ 9 = 27GB of Transmittance images, 3 $\times$ 9 = 27GB of Retardation images, and 3 $\times$ 9 = 27GB of Direction images.

For the Deep Learning model training, we have divided the total 9 images into 3 subsets of 5 images as Training Set, 2 images as Validation Set, and other 2 images as Testing Set.

3.1.2. Developing Dataset
Polarized Light Image:
A small but equivalent (to the large dataset) subset was generated for the developing purposes. Since the main objective of predicting cytoarchitectonic images is to train the neural network on how to distinguish between different types of cells and the border regions, a total of 5 patches of the dimension (4096, 4096, 9) were picked from a full 3D-PLI section including information about all types of brain cells for the experimental dataset.
Each patch of 3D-PLI images was about 1.5 GB in memory size. Hence, the total size of the experimental 3D-PLI sections was approx. 1.5×5=7.5 GB.

Cytoarchitectonic Image:
For the training using the 5 3D-PLI patches, corresponding Cytoarchitectonic pictures of (4096 × 4096) dimension have also been used. Each patch takes up roughly 30 MB of memory. As a result, cytoarchitectonic data was roughly 30 × 5 = 150 MB in total.

Additional Parameterized Map Images and Masks :
The 3D-PLI patches were taken in such a way that they did not contain any background for the experimental training. For the experimental training, there were no mask pictures.
Each image of the Transmittance, Retardation, and Direction maps took up around 86MB of memory. For each map category of Transmittance, Retardation, and Direction map image, we had 86×5=430MB of image data.

For the Deep Learning model training, we have divided

the total 5 patches into 2 subsets of 4 patches as Training Set, and 1 image as Validation Set. We have decided to use the Testing set of the main Large data to test the trained model.

3.2. Data Pre-processing

3.2.1. Calculating The Transmittance, Retardation, and Retardation Map Images

Using Axer et al. (2011)'s approach, we initially only acquire the rotation angle images (a total of 9 images in our case) for each brain section. The raw 3D-PLI data can be characterized using equations (1) and (2). The raw data was then processed to discrete harmonic Fourier analysis, which transferred it from the spatial domain to the Fourier domain (equation (3)), and the Fourier coefficients were estimated using equation (4) and (5). Then, using equation (6), (7), and (8), we computed the Transmittance map, Retardation map, and Direction map images respectively from these raw 3D-PLI data converted to the Fourier domain.

3.2.2. Data Normalization

Since the intensity ranges of various modalities differ. Normalization is therefore essential, both to ensure that the model assigns equal weights to each variable at the start of the training and to lessen the computing load during neural network training. As a result, no single variable may influence the model's learning in a certain way.

We identified the maximum and lowest intensity values for each of the raw rotation angle images for all 9 rotation angles and over all 9 images in the dataset. Then we used min-max normalization to rescale all of the intensity values in the [0, 1] range while maintaining their inter-image intensity ratio.

We only have one image per section for the cytoarchitectonic data. So, for these images, we used min-max normalization to scale the intensity level in the range of [0, 1] while maintaining the inter-image intensity level ratio by determining the maximum and minimum values across all 9 cytoarchitectonic images.

During the calculating procedure, the images of the Transmittance , Retardation and Direction map were already normalized. As a result, no further normalization was required for these images.

3.2.3. Hybrid data Creation from 3D-PLI, Transmittance map, Retardation map, and Direction map for testing Redundancy

The Transmittance, Retardation, and Direction maps in the 3D-PLI images were generated or calculated from the 9 rotation angle images. As a consequence, it appears that including them in deep neural network training is redundant. Nonetheless, we performed a study to see the effect of including the maps in addition to the 9-channel calibrated PLI image.

As a result, we've tried three different strategies by combining these rotation angle images with various maps. To train the deep learning network, the first strategy was to use only 9 rotation angle images. Then we stacked the Retardation map images on top of them. As a result, each section's image dimension became (# of rows of pixels, # of columns of pixels, 10). We also stacked the Direction map images to the rotation and Retardation map images as a third technique, changing the dimension to (#of rows of pixels, #of columns of pixels, 11). Despite the fact that it appeared to be redundant at first, the trained models produced diverse outputs.

3.2.4. Registration of 3D-PLI and Cytoarchiteconic Dataset

Despite the fact that we have 3D-PLI and cytoarchitectonic images as a pair, they are not aligned as a result of different scanning procedures. Therefore, the data was not immediately ready for performing domain transfer. It was also necessary to apply image registration in order to align the 3D-PLI and cytoarchitectonic images.

To register the cytoarchitectonic image to the 3D-PLI image, we used a landmark-based affine transformation. We used a GUI-based application called "Cyto Tilt" developed at Forschungszentrum Jülich to obtain landmarks in both 3D-PLI and cytoarchitectonic pictures. For all the image pairs in the dataset, we have annotated at least 250 landmarks for 3D-PLI and cytoarchitectonic images.

We utilized the OpenCV package in Python to generate the Homography matrix and perform the warping after noting the landmarks and their corresponding coordinated values as shown in Figure 9.

Performing the warping, we got the results presented in Figure 11. The same procedure was followed for all the images in the dataset.

3.2.5. Data Augmentation

We opted to train the model using (256, 256) pixel patches to ease the computation of the U-net model. As a result, we've taken a larger crop with a ratio of $\sqrt{2}$,
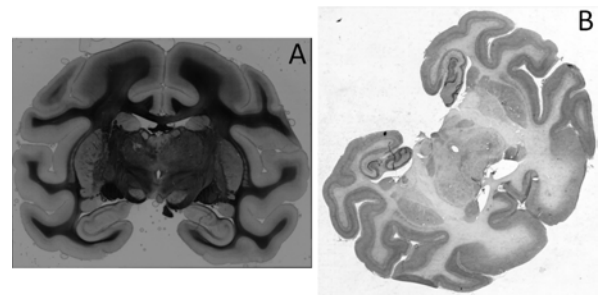
Figure 8: A brain section pair (Section-539) of (A) Transmittance map of the 3D-PLI image, and (B) unregistered cytoarchitectonic image from the dataset.
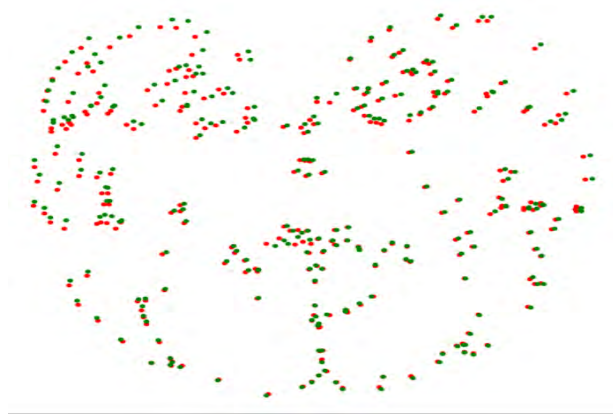
Figure 9: The registration of the landmarks. The red dots are for the fixed 3D-PLI image, and the green dots are for the cytoarchitectonic images after performing the registration.

Map Image		Network	Loss Function		
Transmittance	Transmittance +Retardation	Resnet18	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	Resnet34	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	Resnet152	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	Densenet121	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	Densenet161	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	InceptionV4	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	EfficientB0	L1	MSE(L2)	RMI Loss
Transmittance	Transmittance +Retardation	EfficientB4	L1	MSE(L2)	RMI Loss

Table 1: Appled different approaches to find the optimum working U-net model.

yelding pathces of (362, 362) for each arbitrary patch to ensure that no information is lost throughout the augmentation process.

We used the Albumentation.ai package built by Buslaev et al. (2020) for data augmentation. To augment our dataset, we employed the procedures listed below.

- Affine Rotation, performed for all the images(100%), It performs the affine transformation on the images to rotate them from -180° to 180° randomly with -30° to 30° random deformation preserving translation percentage within 10% in both column and rows.

- Center Crop, performed for all the images(100%), Scales the images uniformly (keep the aspect ratio) so that both of its dimensions (width and height) are equal to or greater than the view's corresponding dimension (minus padding). After that, the image is centered in the view.

- Horizontal Flip, performed for 50% of the images. It flips the images Horizontally.

- Vertical Flip, performed for 50% of the images. It flips the image Vertically

3.3. Computing Resources

3.3.1. JURECA (for Training on Developing Data)

We used the Jülich Research on Exascale Architectures (JURECA) supercomputing facility to train the deep learning model with experimental data. We used one NVIDIA Quadro RTX 8000 GPU with 46080 MB of RAM, clock speeds ranging from 1395 to 14000 MHz, and a bandwidth of 672.0 GB/s.

3.3.2. JUWELS Booster (for Training on Original Data)

We used the Jülich Wizard for European Leadership Science (JUWELS) supercomputing facility to train the deep learning model with original data. We employed four NVIDIA A100 Tensor Core GPUs with 40960MB of memory, a clock speed of 1095 MHz to 1410 MHz, and a bandwidth of 1,555 GB/s apiece for large data training. After parallel processing of the neural network model over all four GPUs, the total compute capability rose fourfold.

3.4. Modality Transfer

We have employed PyTorch developed by Paszke et al. (2019) with the wrap-up libraries of PyTorch Lightning built by Falcon et al. (2020), which are also based on vanilla PyTorch libraries, for our deep learning model training.

3.4.1. U-net based Autoencoder

U-net-based autoencoder models were our first choice for transferring the modality. We used Yakubovskiy (2020)'s library to make implementing various types of deep learning pre-trained models at the encoder path as simple as possible.

The model is depicted in Figure 12

We used a variety of map images as inputs, networks in the encoder path, and Loss functions to discover the highest performing pre-trained model in the U-net architecture, as shown in the Table. 1.

Figure 10: A typical brain section pair of (A) 3D-PLI image, and (B) unregistered cytoarchitectonic image from the dataset with annotated landmarks(The red dots in the 3D-PLI image and the blue dots in the cytoarchitectonic image.).
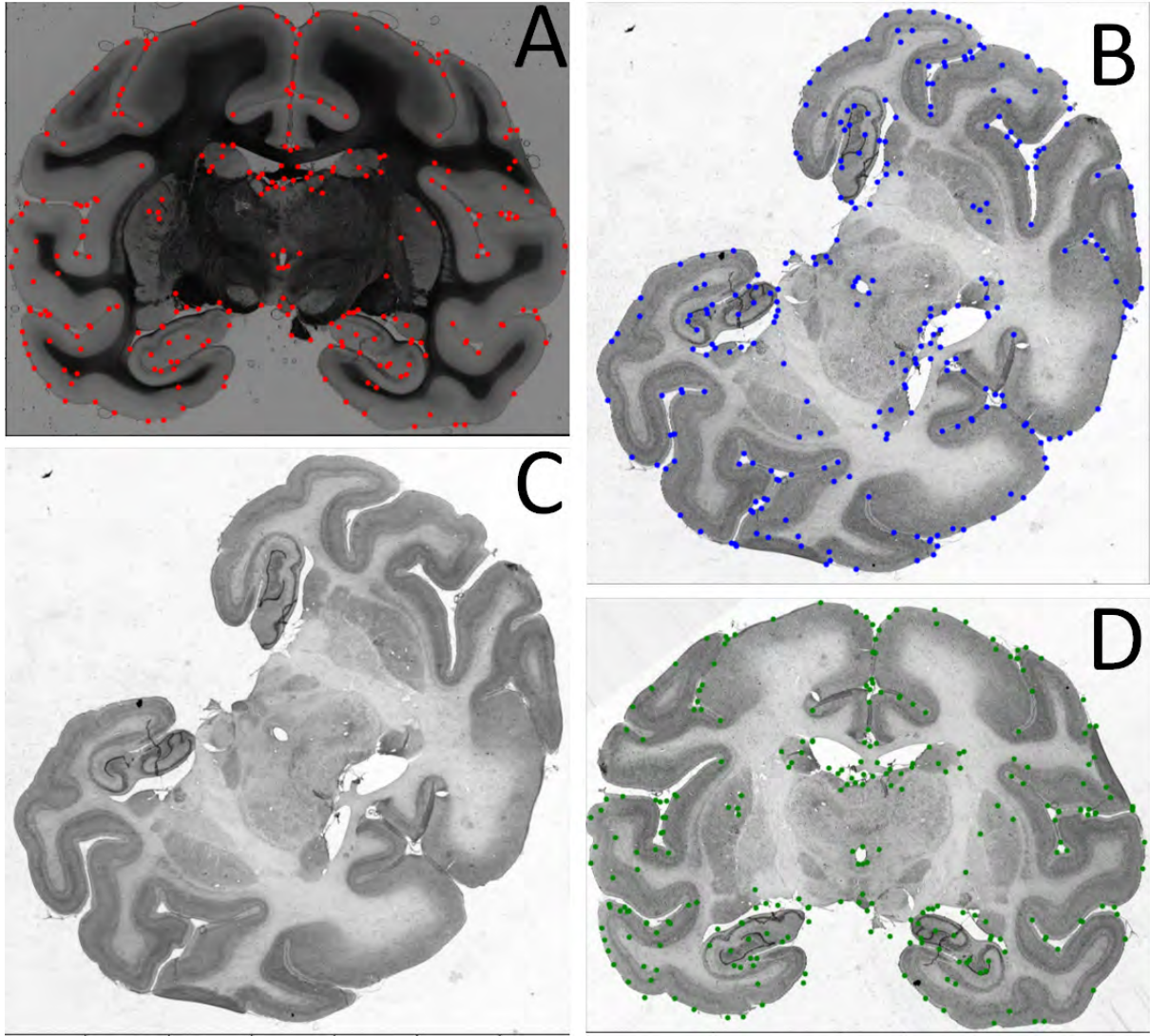
Figure 11: The warping stages of the 3D-PLI and Cyto pair. Image (A) is the 3D-PLI image with landmarks (red dots), image (B) is the cytoarchitenctonic image before the warping with landmarks (blue dots), image (C) is the cytoarchitectonic image without the landmarks, and image (D) is the registered(warped) cytoarchitectonic image.

Therefore, in total we have ran 2X8X3=48 U-net model training with all the possible combinations. For our training, the resnet18 pre-trained model in the encoder with Transmittance + Retardation map images as input image in pair with cytoarchitectonic images and
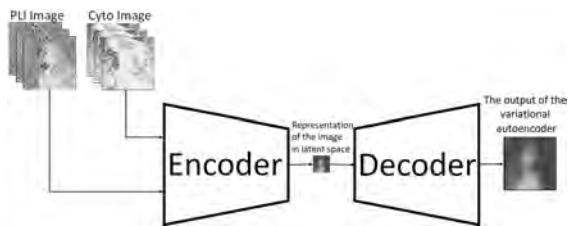
Figure 12: The diagram for the network we intended to use for Variational Auto encoder training. The inputs were 3D-PLI images, and the cytoarchitectonic images. At the encoder path we have used different pre-trained Deep Learning models.

the Region Mutual Information(RMI) Loss worked best. Hence, for initiating the adversarial loss to implement the Conditional Generative Adversarial Network, we used this RMI loss and Resnet18 architecture.

3.4.2. Using Region Mutual Information(RMI) Loss

The dependencies between pixels in an image are ignored by pixel-wise loss algorithms like L1 or MSE. This pixel-wise loss computation method produces unsatisfactory results because our datasets are not precisely aligned and come from different modalities. As a result, we started the Zhao et al. (2019)-described region mutual information loss. Instead of using a single pixel to calculate the loss, an estimating kernel uses a collection of pixels as a single quantity, as shown in the Figure 13.

Then, the entropy and variance are calculated for each

group and compared. The calculation of the Region Mutual Information Loss is based on this principle.

In practice, perfect calculation of mutual information is nearly impossible as they flactuate a lot across the different groups. They first downsampled the image and then calculated the mutual information to maximize their similarity to avoid the loss of variation.

3.4.3. Conditional GAN with RMI Loss Function(Initiating Adversarial Loss)

To improve the output of the U-net, we decided to incorporate adversarial loss, hence an adversarial network was established. We used the previously generated U-net model as the Generator and a predesigned Pytorch lightning-bolt classifier network as the Discriminator in this new Generative Adversarial Network setup. We can control the discriminator network's smartness by altering a hyperparameter called "feature maps" in this Discriminator setup. As the value of "feature maps" rises,

so does the discriminator network's intelligence. Furthermore, we employed the RMI loss function and a regularization parameter RMI_λ to control the impact of the conditional GAN configuration on the training. As a result, we may compare our work to Isola et al. (2017)'s work by comparing Figure 7 and Figure 14. Instead of using noise, we define the conditional GAN using the U-net generated image as input and the RMI loss function as our conditions (cGAN).

Conditions(RMI Loss and Adversarial Loss) to train the Generator and Discriminator

In our deep learning model, we used U-net generated images in place of noise(z), the RMI Loss as our conditioning parameter with the RMIlambda parameter to adjust its impact on the training(x), and the output image(y) is our Cyto image, as compared to the way provided in Isola et al. (2017). When we compare our parameters to equation (9), we get the following,

$$\mathcal{L}_{cGAN}(G, D) = E_{RMI_\lambda*RMILoss, CytoImage}[\log D(RMI_\lambda * RMILoss, CytoImage)]$$
$$+E_{RMI_\lambda*RMILoss, GeneratedImage}[\log(1 - D(RMI_\lambda * RMILoss, G(RMI_\lambda * RMILoss, GeneratedImage)))] \quad (12)$$

As a result, when compared to equation(11), our experiment's purpose is as follows:

$$G = arg \min_{G} \max_{D} \mathcal{L}_{cGAN}(G, D) + \lambda\mathcal{L}_{RMI_\lambda*RMILoss}(G) \quad (13)$$

As a result, in order to attain the optimum performance according to equation(13), we explored a variety of ways by altering the RMI_λ and feature map parameters to govern the network for the best results. The different parameterized training approaches is described in Table. 2.

With RMI_λ=1.0 and Feature Map=64, the best performing outcome was observed after training the models with different 6X5 = 30 parameter setup.

Figure 14: The illustration of the final deep learning model.

4. Results

4.1. U-net based Autoencoder

The results we obtained from the U-net based autoender is shown in Figure 15

Figure 13: A typical multi-dimensional point that corresponds to an image region. An image can be converted into a multi-dimensional distribution of multiple high-dimensional points that encode the relationship between pixels using the same technique.

Generator					Discriminator	
RMIlambda				Trainable Parameters	Feature Map	Trainable Parameters
0.0;	0.2;	0.4;	0.6;	0.8;	1.0;	14.3M	8	44.4K
0.0;	0.2;	0.4;	0.6;	0.8;	1.0;	14.3M	16	174K
0.0;	0.2;	0.4;	0.6;	0.8;	1.0;	14.3M	32	693K
0.0;	0.2;	0.4;	0.6;	0.8;	1.0;	14.3M	34	2.8M
0.0;	0.2;	0.4;	0.6;	0.8;	1.0;	14.3M	128	11M

Table 2: The different parameters used to find the best performing deep learning model.
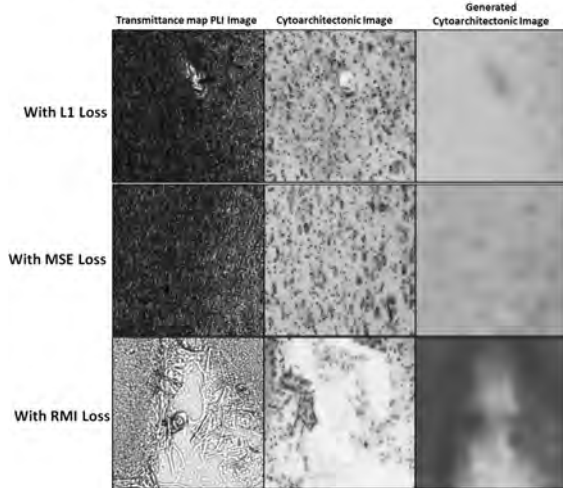
Figure 15: Random patch sets: Output of the U-net autoencoder with resnet18 as encoder, and using L1, MSE, and RMI Loss. The RMI was the best performing one. Hence, we used this RMI loss for further training with adversarial loss.

4.2. Conditional GAN with RMI Loss Function(Initiating Adversarial Loss)

The obtained results from Conditional Generative Adversarial Network with RMI Loss function is shown in Figure 16

4.3. Evaluation

For the time constrain, we chose to apply the model trained on the developing data to the large dataset's testing set. The outcomes appear to be intriguing and promising.

4.3.1. Evaluation: Qualitative

The testing set of the large dataset have 2 section(coronal) of the vervet monkey's brain. The qualitative results are shown in Figure 17 and 18.

4.3.2. Evaluation: Quantitative

For the Quantitative analysis we have used a few evaluation metrices that have commonly been applied to other image modality transfer.(Armanious et al., 2020) The results for the testing set are shown in Table(3):

5. Discussion

5.0.1. Discussion on Expected and Obtained Progressive Results

U-net based Autoencoder

The basic U-net autoencoder deep learning structure downsamples an input image to a low-dimensional latent distribution and then upsamples it to the original dimension. The image, on the other hand, is deformed from the original and is based on a low-dimensional latent representation. As a result, it contains the image's fundamental data. It actually tries to forecast the nearest low-dimensional distribution that can reflect the input using a loss function. Initially, we used the L1 and MSE(L2) losses to estimate this distribution

Figure 16: The training batch size was8. This the illustration of the patched from the last epoch for Input PLI, Cyto Image and The Generated Cyto Image.
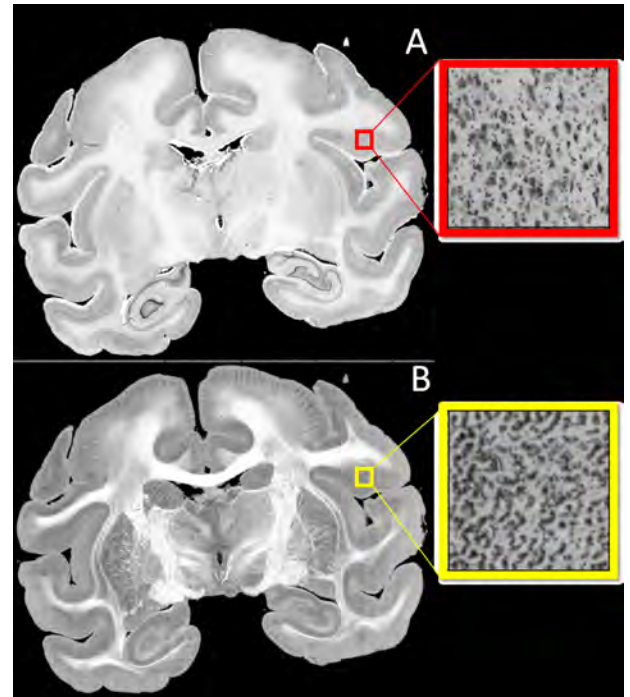
Figure 17: Section No.548. (A) Original Cytoarchitectonic Image and a sample patch, (B) Generated Cytoarchitectonic Image and a sample patch.

8.12

Evaluation Metrics	Section-548	Section-559	Remarks	Reference Ideal Value
Mean Squared Error(MSE)	0.028	0.028	↑↑	≈0
Root Mean Squared Error(RMSE)	0.17	0.17	↑	≈0
Structural Similarity Index (SSIM)	0.44	0.35	—	≈1
Peak Signal to Noise Ratio (PSNR)	63.63	63.57	↑	≈100%
Universal Quality Index(UQI)	0.95	0.94	↑↑	≈1
Erreur Relative Globale Adimensionnelle de Synthesis (ERGAS)	4074	5998	↓↓	≈0
Spatial Correlation Coefficient (SCC)	0.009	0.008	↓↓	≈1
Spectral Angle Mapper(SAM)	0.22	0.29	↑	≈0
Visual Information Fidelity(VIF)	0.15	0.09	↓↓	≈1
Mutual Information	0.72	0.70	↑	≈1
L2 Distance	3327	3289	↓↓	≈0
Average Log-Likelihood	-367M	-354M	↑↑	-∞
Maximum Mean Discrepancy (with Linear Kernel)	173.36	32.12	↑	≈0
Maximum Mean Discrepancy (with Radial Basis Functional Kernel)	0.0001	0.0002	↑↑	≈0

Table 3: Applied different Evaluation metrices' results for Section-548 and Section-559, with a reference value. The Remark column uses the following nomenclature: ↑↑, Near 80-100% of Ideal Value; ↑, Near 60-80% of Ideal value; —, near 40-60% of Ideal Value; ↓↓, Near 20-40% of Ideal Value; and, ↓, Near 0-20% of Ideal Value
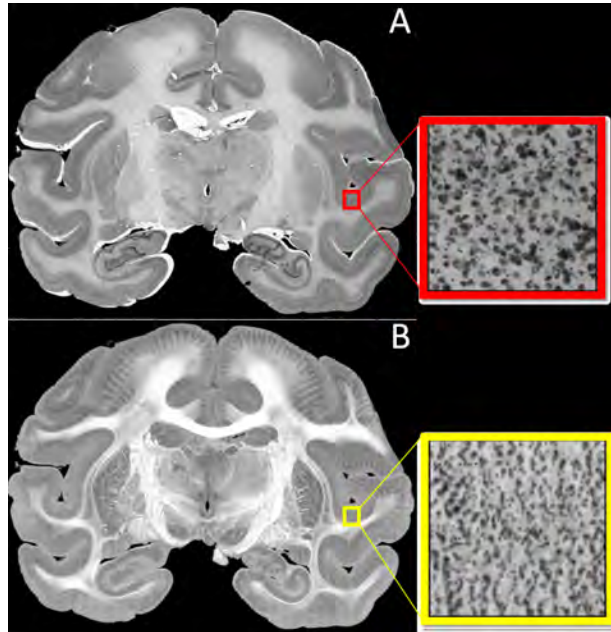
Figure 18: Section No.559. (A) Original Cytoarchitectonic Image and a sample patch, (B) Generated Cytoarchitectonic Image and a sample patch.

in our experiment. Later, we applied the Region Mutual Information RMI Loss function, which has demonstrated promising results by attempting to build a connecting bridge between the PLI and the lower level latent distribution of the Cytoarchitectonic image.

Using Region Mutual Information(RMI) Loss
As previously stated, the information contained in the kernel-sized patches of an image is calculated using a kernel rather than a single pixel in Region Mutual Information(RMI) loss. Figure-14 shows that the RMI Loss may extract more mutual information from the Cytoarchitectonic and 3D-PLI images in our application. We then opted to use this RMI loss as the conditional parameter to create a Conditional Generative Adversarial Network for further training with Adversarial Loss (cGAN).

Conditional GAN with RMI Loss Function(Initiating Adversarial Loss)
The results have been improved significantly after starting adversarial training with RMI loss as the conditional parameter to create a Conditional Generative Adversarial Network (cGAN) and doing a lot of hyperparameter tunning such as regulating RMIlambda, Feature maps, Training Size, and using different metadata as input. Figure 15 shows that the model has learnt the shades of different locations and can also predict cells from 3D-PLI images, as demonstrated by the patches. Although the patches have a wavy shape,

perceptual loss is predicted to remedy the problem. We were unable to add that to our model due to a time constraint. As a result, we'll be working on this project right away to eliminate the wavy artifacts.

5.0.2. *Reliability of the generated Cytoarchitectonic Data*

On the test set of the main large dataset, we ran the Deep Learning model that was trained on small data. The results of calculating the cytoarchitectonic image from the 3D-PLI are displayed in Figure-16(Section-548) and Figure-17(Section-559), as the qualitative analysis. The images are intriguing since the model can accurately predict gray matter and, more importantly, cells in the gray matter region, particularly glial cells. We had 5 patches of a brain section in the experimental limited data, and all of those patches were collected from the gray matter border region of the brain. As a result, this result was obvious, given that the model was meant to learn the gray matter correctly.

There was no information on the subcortical areas in the small training set. As a result, it's understandable that the model's prediction of white matter regions wasn't perfect. As a result, it projected the presence of some cells in the white matter regions (albeit a small number). The model did a good job of predicting L1 regions.

From the evaluation metrices we have used to evaluate the generated image quantitavely, we have received very good accuracy(near to 80-100% of the ideal value) from Mean Squared Error(MSE), Universal Quality Index(UQI), Average Log-Likelihood, and from Maximum Mean Discrepancy(with Radial Kernel). It was expected as in MSE, it consideres the overall mean of the squared value. Similarly according to Wang and Bovik (2002) in UQI, it measures the quality index from the standard deviation of overall distribution of two images. For Average log-likelihood, it measures the likelihood between two images from the difference of their average. If the difference is very small then applying logarithm we get a negative value. As this negative value is close to -∞, the results is good. For our case, we have obtained a big negative number for both the images. And, according to Rabanser et al. (2019) MMD with radial kernel also gives an discrepancy measure for a radial kernel. For Root Mean Squared Error(RMSE), Peak Signal to Noise Ratio(PSNR), Spectral Angle Mapper (SAM), Mutual Information, and MMD with Linear kernel we have obtained good results(60-80)% of the ideal value. According to Wang et al. (2004), PSNR uses the signal noise ratio of two different image by adding noise, and blurriness to measure the similarity. It found better than average similarities in our real cytoarchitectonic and generated cytoarchitectonic images. In Yuhas et al. (1992) they have described a method to find the relative

change in an image of earth taken from space before and after by changing the viewing angle mathematically. In our case, as our images are very large scale image, it can be evaluated with their evaluation. It shows a got perception of recongnizing the same portions from the cytoarchitectonic and generated images. And for the MMD with Linear Kernel as described in Rabanser et al. (2019) it was calculating the discrepancy with the linearly situated neighbouring pixels of the two images. Thus, result was good but less than with the Radial Basisi Functional kernel. With Structural Similarity Index(SSIM), it applies different filtering for performing the evaluation(Wang et al., 2004). SSIM became confused between the evaluation, and have given a nearly confusing result(Neither good, nor bad).

With Erreur Relative Globale Adimensionnelle de Synthesis (ERGAS)described in Renza et al. (2013), Spatial Correlation Coefficient (SCC) as described in Zhou et al. (1998), Visual Information Fidelity(VIF) described in Sheikh and Bovik (2006), and L2 Distance, we have received a bad result as all of them mainly performs pixel to pixel evaluation method. Our Cytoarchitectonic imaging brain sections had some unavoidable distortion in tissues during staining process after 3D-PLI imaging. Hence, those images required registration. We have applied the warping as registration, but still there was error between pixel to pixel positioning in between the 3D-PLI and Cytoarchitectonic images in the testing dataset. So the pixel to pixel evaluation methods were not suitable for our work. Moreover, our model had some wavy shape structure on the predicted images.

5.0.3. *Future Possible Extension of the Work*

The outcome of the project has shown great potential in evaluation results. Next we will perform the training with the large dataset. Also to avoid the wavy structures we will try applying the Perceptual loss as described in Johnson et al. (2016) and Wasserstein loss described in Frogner et al. (2015) to reduce this wavy effect and to increase the performance. Moreover, as pixel to pixel evaluation techniques are not suitable in our case, they idea of Gray Level Index as described in Kiwitz et al. (2020) can be applied for both training and evaluating the project outcome.

6. Conclusions

6.1. *Achievements*

The main goal of this thesis study was to find out if the 3D-PLI image data contains enough information to predict or create synthetic Cytoarchitecture images. Throughout our research we found a potential positive outcome infers that Cytoarchitecnic images can be predicted from transferring the modality of a 3D-PLi image. The resultant images show a feasible visual result.

Furthermore, the evaluation metrics present a promising score. At the end of the present study, we have found that 3D-PLI contains promising amount of latent image information to generate synthetic cytoarchitectonic images.

6.2. Future Work

The future work is intended to use more data and improve the loss functions to improve the results. Our training was completed on the developing data and from out experimentation, we pointed out a few potential approaches such as intiating perceptual loss, wasserstein loss, using the GLI indexing etc. to improve the results.

7. Acknowledgments

Compliance with ethical standards All animal procedures were approved by the institutional animal welfare committee at Forschungszentrum Jülich GmbH, Germany, and were in accordance with the European Union and National Institutes of Health guidelines for the use and care of laboratory animals and in compliance with the ARRIVE guidelines.The monkey brains were obtained in accordance with the Wake Forest Institutional Animal Care and Use Committee (IACUC #A11-219). Euthanasia procedures conformed to the AVMA Guidelines for the Euthanasia of Animals, using ketamine/pentobarbital anesthesia followed by perfusion with phosphate buffered saline and fixation with 4 % paraformaldehyde.

I would like to thank all the coordinators and administrators of my master's study program, Medical Imaging and Applications(MAIA) from the deepest corner of my heart for providing the opportunity to learn and to get prepare myself for master thesis. Also I would like to thank the MAIA authority and European Education and Culture Executive Agency(EACEA) for supporting my education throughout the whole master's study program by the provind the Erasmus Mundus Joint Scholarship.

I would like to thank my supervisors Prof. Dr. Timo DICKSHEID and Prof. Dr. Markus AXER for all the suggestions and guidance they provide for the thesis work. I would also like to thank Mr. Marcel HUYSEGOMS from INM-1 group for providing his Cyto_tilt landmark identification tool for our project. I wish him good luck for his developing project. I also like to thank Dr. Susanne Wenzel, the scientific coordinator or INM-1 group for providing all the administrative support in forschungszentrum jülich.

At last, I want especially thank Mr. Esteban Vaca and Mr. alexander oberstraß, PhD researchers and my supervisors at forschungszentrum jülich for supervising me for the whole thesis period. This challenging thesis could not be finished properly within the limited timeframe without the massive support I received from them 24/7 of the days, both with the conceptual study and programming processes.

References

Armanious, K., Jiang, C., Fischer, M., Küstner, T., Hepp, T., Nikolaou, K., Gatidis, S., Yang, B., 2020. Medgan: Medical image translation using gans. Computerized medical imaging and graphics 79, 101684.

Axer, M., Grässel, D., Kleiner, M., Dammers, J., Dickscheid, T., Reckfort, J., Hütz, T., Eiben, B., Pietrzyk, U., Zilles, K., et al., 2011. High-resolution fiber tract reconstruction in the human brain by means of three-dimensional polarized light imaging. Frontiers in neuroinformatics 5, 34.

Braak, H., 1980. Studies of brain function, vol 4: Architectonics of the human telencephalic cortex.

Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A., 2020. Albumentations: Fast and flexible image augmentations. Information 11. URL: https://www.mdpi.com/2078-2489/11/2/125, doi:10.3390/info11020125.

de Campos Vidal, B., Mello, M.L.S., Caseiro-Filho, A.C., Godo, C., 1980. Anisotropic properties of the myelin sheath. Acta histochemica 66, 32–39.

Falcon, W., Borovec, J., Wälchli, A., Eggert, N., Schock, J., Jordan, J., Skafte, N., Bereznyuk, V., Harris, E., Murrell, T., et al., 2020. Pytorchlightning/pytorch-lightning: 0.7. 6 release. Zenodo: Geneva, Switzerland .

Frogner, C., Zhang, C., Mobahi, H., Araya, M., Poggio, T.A., 2015. Learning with a wasserstein loss. Advances in neural information processing systems 28.

Glazer, A., Lewis, J., Kaminsky, W., 1996. An automatic optical imaging system for birefringent media. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences 452, 2751–2765.

Grau-Moya, J., 2011. Integration of the information in complex neural networks with noise .

Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1125–1134.

Jacques, S., Christe, B., 2020. Chapter 2 - healthcare technology basics, in: Jacques, S., Christe, B. (Eds.), Introduction to Clinical Engineering. Academic Press, pp. 21–50. URL: https://www.sciencedirect.com/science/article/pii/B9780128181034000028, doi:https://doi.org/10.1016/B978-0-12-818103-4.00002-8.

Johnson, J., Alahi, A., Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution, in: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham. pp. 694–711.

Kiwitz, K., Schiffer, C., Spitzer, H., Dickscheid, T., Amunts, K., 2020. Deep learning networks reflect cytoarchitectonic features used in brain mapping. Scientific Reports 10, 1–15.

Krause, D., Thörnig, P., 2018. JURECA: Modular supercomputer at Jülich Supercomputing Centre. Journal of large-scale research facilities 4, 132. doi:10.17815/jlsrf-4-121-1.

Liu, M.Y., Breuel, T., Kautz, J., 2017. Unsupervised image-to-image translation networks. Advances in neural information processing systems 30.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library, in: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Rabanser, S., Günnemann, S., Lipton, Z., 2019. Failing loudly: An empirical study of methods for detecting dataset shift. Advances in Neural Information Processing Systems 32.

Reckfort, J., Wiese, H., Pietrzyk, U., Zilles, K., Amunts, K., Axer, M., 2015. A multiscale approach for the reconstruction of the fiber architecture of the human brain based on 3d-pli. Frontiers in neuroanatomy 9, 118.

Renza, D., Martinez, E., Arquero, A., 2013. A new approach to change detection in multispectral images by means of ergas index. IEEE Geoscience and Remote Sensing Letters 10, 76–80. doi:10.1109/LGRS.2012.2193372.

Schiffer, C., Harmeling, S., Amunts, K., Dickscheid, T., 2021. 2d histology meets 3d topology: Cytoarchitectonic brain mapping with graph neural networks, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 395–404.

Sheikh, H.R., Bovik, A.C., 2006. Image information and visual quality. IEEE Transactions on image processing 15, 430–444.

Shen, D., Wu, G., Suk, H.I., 2017. Deep learning in medical image analysis. Annual review of biomedical engineering 19, 221.

Sorin, V., Barash, Y., Konen, E., Klang, E., 2020. Creating artificial images for radiology applications using generative adversarial networks (gans)–a systematic review. Academic radiology 27, 1175–1185.

Takemura, H., Palomero-Gallagher, N., Axer, M., Gräßel, D., Jorgensen, M.J., Woods, R., Zilles, K., 2020. Anatomy of nerve fiber bundles at micrometer-resolution in the vervet monkey visual system. elife 9, e55444.

Vey, B.L., Gichoya, J.W., Prater, A., Hawkins, C.M., 2019. The role of generative adversarial networks in radiation reduction and artifact correction in medical imaging. Journal of the American College of Radiology 16, 1273–1278.

Wang, Y., Yu, B., Wang, L., Zu, C., Lalush, D.S., Lin, W., Wu, X., Zhou, J., Shen, D., Zhou, L., 2018. 3d conditional generative adversarial networks for high-quality pet image estimation at low dose. Neuroimage 174, 550–562.

Wang, Z., Bovik, A.C., 2002. A universal image quality index. IEEE signal processing letters 9, 81–84.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing 13, 600–612.

Wei, W., Poirion, E., Bodini, B., Durrleman, S., Ayache, N., Stankoff, B., Colliot, O., 2019. Predicting pet-derived demyelination from multimodal mri using sketcher-refiner adversarial training for multiple sclerosis. Medical image analysis 58, 101546.

Wilson, S.M., Bacic, A., 2012. Preparation of plant cells for transmission electron microscopy to optimize immunogold labeling of carbohydrate and protein epitopes. Nature protocols 7, 1716–1727.

Xiang, L., Wang, Q., Nie, D., Zhang, L., Jin, X., Qiao, Y., Shen, D., 2018. Deep embedding convolutional neural network for synthesizing ct image from t1-weighted mr image. Medical image analysis 47, 31–44.

Xie, X., Niu, J., Liu, X., Chen, Z., Tang, S., Yu, S., 2021. A survey on incorporating domain knowledge into deep learning for medical image analysis. Medical Image Analysis 69, 101985.

Yakubovskiy, P., 2020. Segmentation models pytorch. 2020. URL https://github. com/qubvel/segmentation_models. pytorch .

Yuhas, R.H., Goetz, A.F., Boardman, J.W., 1992. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm, in: JPL, Summaries of the Third Annual JPL Airborne Geoscience Workshop. Volume 1: AVIRIS Workshop.

Zeineh, M.M., Palomero-Gallagher, N., Axer, M., Gräßel, D., Goubran, M., Wree, A., Woods, R., Amunts, K., Zilles, K., 2017. Direct visualization and mapping of the spatial course of fiber tracts at microscopic resolution in the human hippocampus. Cerebral cortex 27, 1779–1794.

Zhao, S., Wang, Y., Yang, Z., Cai, D., 2019. Region mutual information loss for semantic segmentation. Advances in Neural Information Processing Systems 32.

Zhou, J., Civco, D.L., Silander, J., 1998. A wavelet transform method to merge landsat tm and spot panchromatic data. International journal of remote sensing 19, 743–757.

Zhou, L., Schaefferkoetter, J.D., Tham, I.W., Huang, G., Yan, J., 2020. Supervised learning with cyclegan for low-dose fdg pet image denoising. Medical image analysis 65, 101770.

Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E., 2017. Toward multimodal image-to-image translation. Advances in neural information processing systems 30.

Zilles, K., Palomero-Gallagher, N., Amunts, K., 2015. Myeloarchitecture and maps of the cerebral cortex, in: Toga, A.W. (Ed.), Brain Mapping. Academic Press, Waltham, pp. 137–156. URL: https://www.sciencedirect.com/science/article/pii/B9780123970251002098, doi:https://doi.org/10.1016/B978-0-12-397025-1.00209-8.
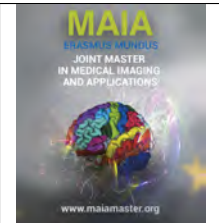
Transfer Learning from Cine to Late Gadolinium Enhancement MRI for Myocardial Segmentation in Patients with Acute Myocardial Infarction

Saud Ahmad Khan[a], Thomas Dietenbeck[a,b], Nadjia Kachenoura[a,b]

[a]*Sorbonne University, INSERM 1146, CNRS 7371, Laboratoire d'Imagerie Biomédicale, Paris, France*
[b]*Institute of Cardiometabolism And Nutrition (ICAN), La Pitié-Salpêtrière Hospital, Paris, France*

Abstract

Cine-bSSFP MRI sequence is widely used in clinical routine for assessing cardiac volumes. Such routine use fostered the development of automated and semi-automated solutions for segmenting Cine MRI sequences. Late Gadolinium Enhancement (LGE) MRI is nowadays a well-established sequence for myocardial scar evaluation in various disease conditions including myocardial infarction. While LGE can successfully highlight the scar tissue, the quantification of its exact size in clinical practice is still a semi-automated process depending on the expertise of the radiologists.. In this study, we present a transfer learning approach from cine toward LGE images for left ventricular myocardial segmentation in the setting of acute myocardial infarction, Convolutional neural networks for regression or segmentation-based approaches are trained on the large-scale databases of Cine MRI and the learning weights of the best model are used to train the same model on LGE MRI data. The methods are evaluated on an LGE-MRI database of 127 patients with whole heart coverage, varying size of myocardial infarction. Our best method delineates the left-ventricular cavity and myocardium with a Dice score of 93.4% ± 6% and 90.0% ± 4% respectively, and was relatively robust to slice position, imaging center as well as infarct size, highlighting its potential usefulness as a promising approach towards segmenting LGE MRI.

Keywords: Cardiac MRI, Cine-bSSFP, Late Gadolinium Enhancement, Myocardial Infarction, Transfer Learning

1. Introduction

Cine MRI images are quite abundant as such sequence is systematically acquired during cardiac MRI exams in clinical routine as well as in clinical research protocols . Such images have high resolution and high contrast and offer a full temporal and spatial coverage since they are acquired at multiple time phases throughout the cardiac cycle, while covering the heart from its base to its apex. The left ventricle is systematically segmented of such MRI images in clinical routine to evaluate heart volume and function through volume-derived indices such as ejection fraction and stroke volume, playing a major role in the diagnosis of heart disease.

Late Gadolinium Enhancement (LGE) MRI sequences, also known as Delayed-Enhancement MRI, are mainly used to detect replacement or dense myocardial fibrosis in various disease settings including : myocardial infarction, hypertrophic cardiomyopathy and myocarditis. These sequences are ECG triggered to be acquired during the diastolic phase, when the heart has a stable volume. A gadolinium-based contrast agent is injected to the patient and the MRI signal is acquired 10 to 20 minutes after injection. LGE MRI enables the delineation of regions of dense fibrosis within the myocardium due to changes in the washout patterns through myocardial regions, with a combination of reduced perfusion and delayed washout in the regions with fibrosis. Amount of LGE has a high prognosis value and allows a strong prediction of heart function recovery after revascularization. Similar to Cine MRI, analyzing LGE MRI images also requires the segmentation the manual delineation of the myocardium prior to the application of clustering or thresholding techniques to segment and quantify the amount of dense fibrosis. (Kachenoura et al., 2008) (Baron et al., 2013)

While manually delineating the ventricles and myocardium in Cine MRI is challenging in itself, it is an even more challenging task in LGE MRI images due to
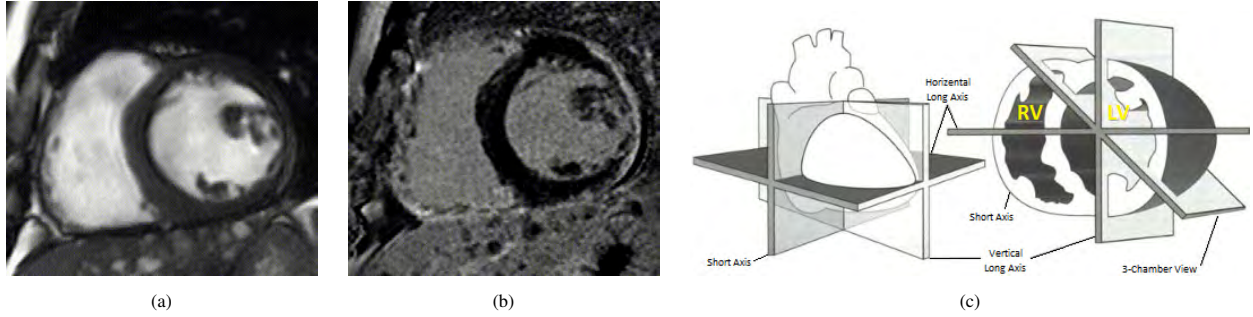
Figure 1: (a) Cine MRI, (b) corresponding LGE MRI and (c) cardiac MRI short-axis view schematic

their lower quality and contrast (especially between LV cavity and the adjacent sub-endocardial scar area).

The large amount of available annotated Cine MRI data allowed for the development of numerous fully automated deep learning-based algorithms, which could be applied on the smaller LGE datasets through transfer learning strategies. Accordingly, the specific aim of this internship was to evaluate the value of such transfer learning strategy from cine MRI towards LGE images for myocardial segmentation in the setting of chronic myocardial infarction.

2. State of the art

2.1. Literature Review

In the existing methods for the delineation of the left-ventricular cavity and the myocardium in LGE MRI sequences, some pertain to segmenting the myocardium as a whole while others focus on scar quantification and segment those regions separately from the rest of the healthy myocardium. Since our focus is on segmenting the whole myocardium region, we will focus on the methods for the former approach more. Indeed, once, myocardium is segmented from LGE images, clustering can be applied to further quantify the infarct volume. (Kachenoura et al., 2008)(Baron et al., 2013)

In the context of deep learning, there has been many attempts so far on using some form of domain adaptation from Cine and T2 sequences towards the LGE MRI. In fact, there are only sparing previous works using a dataset based only on the LGE MRI sequences for the segmentation of the whole myocardium. (Yue et al., 2019) used a shape reconstruction and spatial prior constraints-based network (SRSCN) on LGE MRI from 45 patients resulting in a Dice score of 75.8% for myocardial segmentation. Most of the deep learning-based domain adaption work for the LGE MRI on the delineation of the whole myocardium region comes from the STACOM MS-CMRSeg 2019 challenge (Zhuang et al., 2020). The challenge dataset has Cardiac MRI for 45 patients in the Cine(bSSFP), T2-weighted and LGE MRI images. In the training dataset, Cine and T2-sequences are provided from all the patients (with

the last ten not being labeled). Labeled LGE MRI images, however, come only from 5 patients for validation purpose. The results for this challenge are listed in Table **??**. The design of the challenge encourages domain adaption. The participating teams of the challenge mostly have a two-step solution where the first step uses some supervised or unsupervised form of augmentative and generative technique while the second step focuses on a segmentation network. LGE stylized images are fundamentally synthesized from the other two provided modalities focusing on multi-modal image to image translation, two parameter-sharing segmentation networks and classical image processing techniques to change the image features of the source modalities to the target modality i.e LGE MRI sequences. The average results from the selected methods on the challenge are Dice scores of 89.1% and 76.6% on the left-ventricular cavity and myocardium, respectively.

One might highlight (Vesal et al., 2019) approach which is the only participating team using the mainstream transfer learning technique of deep learning where the learning weights of one model are used to boost a second model that is being trained on a different dataset. In their implementation, they train one U-Net based model on Cine and T2w MRI sequences. Once a model with good performance is achieved, a second U-Net based model using LGE MRI sequences from the four patients in training dataset is trained while being initialized with the learning weights of the first model. For their validation data, they used the LGE MRI sequences from the remaining patient out of the five and get Dice scores of 87.1% and 74.9% of dice on the left-ventricular cavity and myocardium, respectively. On the test dataset from the challenge, however, their performance goes up to Dice scores of 91.2% and 78.9% on the left-ventricular cavity and myocardium. Their results clearly show an improvement in performance using the transfer learning technique.

The MICCAI EMIDEC 2020 challenge (Lalande et al., 2021) is also worth mentioning here. For the dataset for this challenge, there are LGE MRI images from 150 patients with one third of the patients having myocardial infarction and no reflow. The dataset is fur-

Reference	Method	Dice Score (%)		Hausdorff Distance (mm)	
		Left Ventricle	Myocardium	Left Ventricle	Myocardium
Yue et al. (2019)	Shape Reconstruction and Spatial Priors Constraints U-Net (SCSRN)	91.5 ± 5.2	81.2 ± 10.5	11.04 ± 5.818	12.25 ± 6.455
Chen et al. (2019a)	Multimodal Unsupervised Image-to-Image Translation + Cascaded U-Net	91.9 ± 2.6	82.6 ± 3.50	10.28 ± 3.376	12.45 ± 3.142
Wang et al. (2019a)	Attention U-Net with Group-Wise Feature Re-Calibration Module	89.6 ± 4.7	79.6 ± 5.90	13.59 ± 5.206	15.70 ± 5.814
Ly et al. (2019)	Threshold Connection Layer Network (TCL-Net)	87.0 ± 5.1	70.5 ± 11.5	41.74 ± 7.696	42.79 ± 13.26
Wang et al. (2019b)	Selective Kernet U-Net (SK-UNet)	92.6 ± 2.8	84.3 ± 4.80	9.748 ± 3.280	11.65 ± 4.002
Campello et al. (2019)	CycleGAN + U-Net	89.8 ± 4.5	81.0 ± 6.10	10.78 ± 4.066	11.96 ± 3.620
Vesal et al. (2019)	U-Net with Trasnfer Learning	91.2 ± 3.4	78.9 ± 7.30	11.29 ± 4.569	12.54 ± 3.379
Roth et al. (2019)	Multi-Atlas + Anisotropic Hybrid Network (AH-NET)	89.9 ± 4.3	78.0 ± 4.70	11.58 ± 7.524	16.25 ± 6.336
Liu et al. (2019)	Residual U-Net	88.4 ± 0.7	75.1 ± 11.9	14.30 ± 8.170	14.75 ± 7.823
Chen et al. (2019b)	U-Net with Discriminator	82.4 ± 6.8	61.0 ± 10.2	23.69 ± 14.66	24.62 ± 12.66

Table 1: Results on Whole Myocardium Segmentation

Reference	Method	Dice Score (%)			
		Myocardium	Infarction	NoReflow	Average
Zhang (2020)	2D U-Net Variant + 3D U-Net Variant	87.9 ± 2.70	71.2 ± 26.8	78.5 ± 39.3	79.2
Feng et al. (2020)	2D U-Net with Dilated Convolution	83.6 ± 12.4	54.7 ± 34.0	72.2 ± 43.2	70.2
Yang and Wang (2020)	2D U-Net with SE and SK blocks	85.5 ± 2.70	62.8 ± 31.5	61.0 ± 46.3	69.8
Hüllebrand et al. (2020)	2D U-Net Variant + Mixture Models	84.1 ± 5.10	37.9 ± 29.6	52.3 ± 48.3	58.1
Camarasa et al. (2020)	3D U-Net Variant	75.7 ± 11.1	30.8 ± 28.0	60.5 ± 48.5	55.7
Zhou et al. (2020)	2D U-Net with Attention	82.5 ± 5.70	37.8 ± 30.9	52.0 ± 48.7	57.4
Brahim et al. (2020)	2D U-Net with Attention and IRB + 3D U-Net Variant	79.1 ± 5.00	26.4 ± 37.9	64.1 ± 47.9	56.5
Girum et al. (2020)	2D U-Net with SE Block	80.3 ± 5.70	35.0 ± 47.4	78.0 ± 41.4	64.4
Brahim et al. (2022)	Inclusion and Classification Prior InformationU-Net (ICPIU-Net)	87.6 ± ?.??	73.3 ± ?.??	81.3 ± ?.??	80.8

Table 2: Results from Literature on the EMIDEC2020 Dataset

ther divided into 100 patients for training dataset and 50 patients for testing dataset representing equal proportions of healthy and diseased patients. The challenge is targeted for the segmentation of the myocardium into three classes including healthy, infarcted and no-reflow regions. While, it has been mentioned that some participants segmented the myocardium and left-ventricular region first before segmenting the infarcted and no-reflow regions, no specific algorithms and results for such a task are given. This renders the comparison of our findings against the EMIDEC challenge not very straightforward, unless if we consider that the entire myocardium is a union of all the above mentioned three classes. Table 2 gives the results on the EMIDEC testing dataset along with the methods used by the teams. Due to the relatively large training dataset this time, the methods listed here do not employ any use of domain adaptation or transfer learning and focus on the segmentation models. The segmentation models being used are all based on U-Net architecture and its variants.

With the STACOM MS-CMRSeg 2019 challenge, we can see that these methods are clearly plagued by the lack of a proper and labeled LGE MRI dataset. It also seems that a generic deep-learning based solution for segmentation of the whole myocardium and the left-ventricular in the LGE MRI is lacking as the methods mentioned here are working with the Cine and T2w modalities to extract the solution for the corresponding LGE modality. While MICCAI EMIDEC 2020 challenge does solve the issue regarding the availability of relatively larger datasets on the LGE MRI sequences, we believe evaluating robustness of algorithms on LGE data acquired in different centers and different MRI scanner is still needed in the present literature. While

the challenge article(Lalande et al., 2021) does mention that the one third ratio of healthy to diseased myocardium as seen in the challenge dataset is analogous to what is seen in real life, we believe that larger datasets with varying amounts of infarcted myocardium can be beneficial for improving the deep-learning based methods. Indeed, each patients had anyway few slices free of myocardial infection offering the algorithm a sufficient domain to learn how to manage LGE free images. The idea of domain translation that is lacking in the methods used for the EMIDEC 2020 challenge also seems a promising aspect to explore even with relatively larger LGE MRI dataset now being available. Lastly, a contour regression-based deep learning approach in comparison with segmentation-based approach is also a possible venue to explore as such implementations would allow gain in computation power and processing time but are still lacking in literature.

2.2. Contribution

In this study, we present a method for the delineation of left-ventricular cavity and myocardium in the LGE MRI images based on the transfer learning technique used in deep learning. We train and evaluate our meth- ods on a diverse LGE MRI dataset which comes from multiple vendors and is acquired using MRI scan- ners from Siemens, General Electric as well as Philips. Our dataset features 127 patients who have varying level of myocardial infarction, with 12 slices per patient on average. In sum- mary, we observe the effects of deep learning-based transfer learning from Cine to LGE MRI sequences. We also compared the performance of regression-based meth- ods against 2D and 3D segmentation-based methods. Within segmentation-
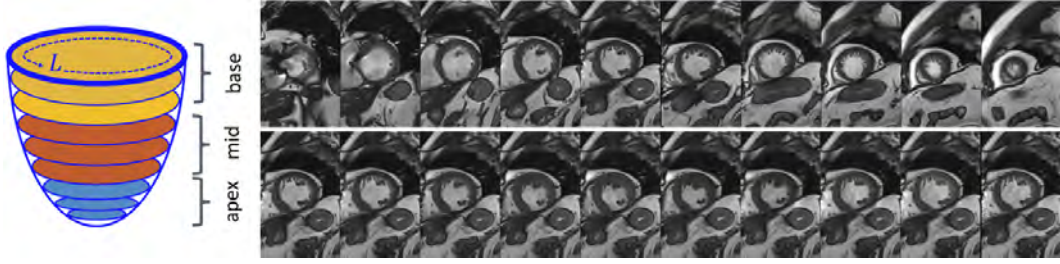
Figure 2: (clockwise) LV schematic from basal to apex, (a) stack of cine short axis images covering the heart from base to apex, (b) mid left ventricular slice throughout the cardiac cycle

(a) Cine MRI Dataset				
Dataset	**Patients**	**Slices**	**Spatial Depth**	**Temporal Depth**	**Pathology Remarks**
	71	10396			Healthy
	30	2245	One Basal		CVD
Private	61	12471	One Mid-Ventricular	Full	
	116	9061	One Apical		HCM
	208	24465			MI
ACDC2017	100	1808	Full	ES & ED	Healthy + MI + DCM + RV + HCM
LVQuan2018	145	2900	Middle		MI + DCM + HCM
(b) LGE MRI Dataset				
Private (24 Centers)	127	1511*	Full	ED	MI

Table 3: Description of available data from (a) Cine and (b) LGE MRI sequences in terms of number of patients and number of annotated images [*1511 = 427 Basal + 640 Middle + 444 Apical Slices].
ED & ES: End Diastole and Systole, CVD: Cardiovascular Disease, MI: Myocardial Infarction, RV: Right Ventricular Cardiomyopathy DCM: Dilated Cardiomyopathy, HCM: Hypertrophic Cardiomyopathy

based methods, we explore different architectures from standard U-Net to more recent transformer-based hybrid U-Net and report our findings.

3. Material and methods

3.1. Dataset

The dataset used in this study pertains to two different cardiac MRI sequences, namely Cine MRI and Late Gadolinium Enhancement (LGE) MRI. Both datasets comprised subjects coming from local private datasets (constituted at Laboratoire d'Imagerie Biomédicale (LIB) in collaboration with European Hospital Georges Pompidou and The Pitié-Salpêtrière Hospital) or from publicly available datasets. Of note, patients in the private datasets were acquired within clinical research protocols approved by the local ethics committee.

Cine MRI datasets had two types of depth: a spatial depth to cover the heart from its base to its apex, Figure 2(a); a temporal depth, since for each of these slices, the heart is imaged throughout the cardiac cycle Figure 2(b). For all Cine MRI images coming from the local private datasets, annotations cover the complete temporal depth (entire cardiac cycle) on one basal, one mid-ventricular and one apical slice in terms of spatial depth. This can be explained by the fact that these patients were analyzed in the setting of strain analysis using a feature tracking algorithm, restricting the

functional estimates to 3 representative slices in compliance with the left ventricular American Heart Association segmentation. The publicly available datasets (ACDC2017 (Bernard et al., 2018) and LVQuan2018 (Xue et al., 2021) challenges) only had annotations on extreme phases (end diastolic and end systole phases) in terms of temporal depth, in compliance with volumes and ejection fraction estimation. The ACDC2017 dataset has complete spatial depth whereas we only have the mid-ventricular slices for LVQuan2018. To summarize, we had about 63,346 images from about 631 patients for cine MRI Table 3(a).

LGE MRI only had a spatial depth since LGE acquisition is performed during diastasis, when the motion of the relaxing heart is minor. Indeed, these images are acquired 10 to 15 minutes after injection of Gadolinium contrast agent to capture areas of myocardial scar and micro-vascular obstructions. We therefore apply a full coverage of the left ventricle from its base to its apex, while positioning continuous short axis slices. In terms of quantity, we have about 1511 images from 127 patients coming from a local private dataset Table 3(b). Accordingly, one might note that the Cine MRI dataset is forty times larger as compared with the LGE MRI dataset.

In the private database, LGE left ventricular myocardial contours were all traced manually by an experienced radiology technician while varying and tuning intensity windowing while Cine MRI images were ana-

lyzed using a custom feature tracking software (Lamy et al., 2018) with visual supervision of the heart contours delineation. Our dataset for LGE MRI sequences also had the diseased regions in the myocardium annotated but due to our task being focused on segmenting the myocardium region as a whole, we only used them for analyzing the quality of our results in varying ratio of infarcted to healthy regions as will be seen in the results section. For further details on both the Cine and LGE MRI datasets, please refer to Table 3

For the regression-based method, we have used a contour extraction algorithm to get the contours of the left-ventricular endocardium and epicardium in case of the publicly available datasets used as we only have the ground truth masks for the regions of interest instead of a contour. More details on this are present in Section 3.2 on regression-based methods.

In order to have a robust model and to fine-tune the hyperparameters of the model, both datasets were split into three parts in proportions of 70-20-10% for training, validation and testing datasets, respectively. To make these splits, features related to the dataset such as the pathology of the patient, the clinical centers and the MRI machines that the data were acquired on are accounted for in equivalent proportions for each split. It allows for a good representation of all the diverse features of the data to be learned by the model. Once the dataset was split, we used 442, 126 and 63 patients for training, validation and testing, respectively, in Cine MRI dataset. In LGE MRI dataset, we use 88, 25 and 12 patients for training, validation and testing, respectively.

3.2. Regression-Based Method

In this study, we tested both regression-based and segmentation-based methods. Although the focus has been on segmentation-based methods, for a more natural and intuitive flow, we will start by describing the regression-based method that have been used in the present work. For this method, Cardiac MRI images are given as input data whereas the target outputs are the left-ventricular myocardial contours, or more precisely, the endocardial and epicardial contours. We used boundary points (Du et al., 2018) representation for the contours in order to prepare a target vector for this regression-based approach that can be processed by the model.

3.2.1. Boundary Points Representation

The myocardial boundaries are represented by a set of discrete points and these points are described by their coordinates:

$$v_i = (x_i, y_i)|_{(i=1,...,n)} \tag{1}$$

where v_i is the ith boundary point's coordinate and n is the number of discrete points of each boundary which depends on the interval between every two adjacent points. Smaller interval sizes lead to bigger n and provide more realistic and smoother representation of the boundary, with a higher precision. The discrete boundary points are obtained using the spline method. The first point intersecting the boundary and horizontal center line is taken and the remaining n-1 points are sampled clockwise along the boundary, evenly and successively. The process is repeated for each boundary (left ventricular endocardium and epicardium in our case). Since we are in discrete domain already, we use the full pixel resolution available to us for each contour. However, since every image has its own extent of epicardium and endocardium contour, we first find the maximum and minimum limit of both the coordinates of the epicardium and endocardium contours and do padding. Figure 5 shows a straight line towards the end of each of the four plots corresponding to the contours of the epicardium and the endocardium depicted in the graph.

Since the left-ventricular cavity and myocardium can have unique shape variations specific to each patient and pathology and the acquisition source of the MRI, the boundary points representation provides a more robust structure compared with other conventional methods like PCA shape (Cootes et al., 1995). Indeed compared to PCA, thes representation avoids a learning step and the need to build a training set distinct from the one used for the DL training.

3.2.2. Architecture

To achieve this regression task, we used standard deep convolutional neural network models such as VGG, ResNet and DenseNet. While these networks are normally used for classification-based tasks, we can extend their functionality for regression-based tasks, by modifying the final classification layer from the number of classes to be predicted, into the length of the target vector representing the boundary points for the contours to be predicted. While we experimented the different variants of the aforementioned architectures, for the purpose of illustration, we can see the default VGG architecture in Figure 3 to briefly summarize how the model works.

3.2.3. Transfer Learning in Regression-Based Method

The networks discussed in the previous section can all be used in their pre=trained form i.e using the weights these models learned for the ImageNet (Deng et al., 2009) dataset challenge. Since we have already mentioned that we change the last layer to represent the length of our target boundary vector, this is already a form of transfer learning. Although the ImageNet dataset is not similar to the cardiac MRI dataset, the weights corresponding to the learning of low-level features like boundary and edge detection remain relevant. To use a pre-trained model with a new dataset, several options can be employed. One of these options consists in modifying a single last layer for our desired out-
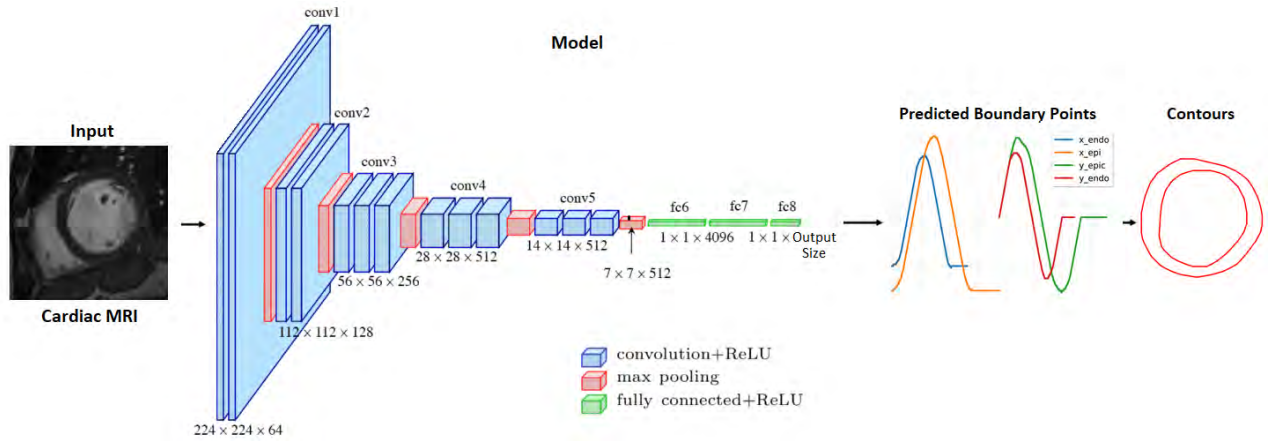
Figure 3: VGG19 architecture with output layer modified for our regression task (Simonyan and Zisserman, 2015)
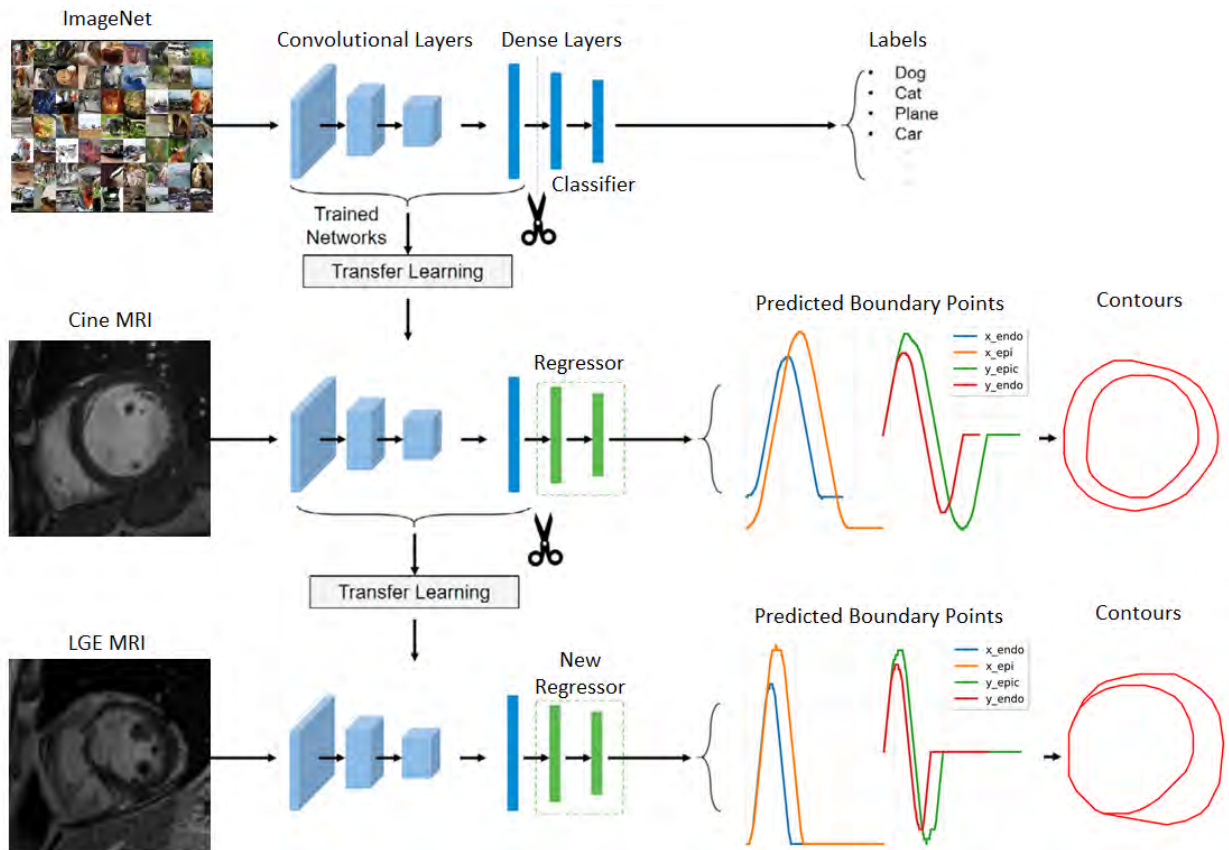
Figure 4: Step-by-step depiction of transfer learning starting from a pre-trained CNN model with imagenet weights, converting the last layer for regression and retraining on the source (Cine MRI) dataset. In the last step, the last layer is again converted to suit the target output size of the target dataset i.e LGE MRI and the model is retrained again with weights from the model trained on Cine MRI

9.6
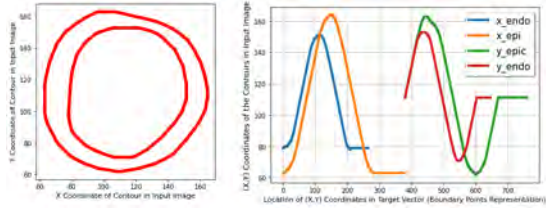
Figure 5: Getting Boundary Points Representation from Contours. The x-axis in Boundary Points Representation is the location of the x and y coordinates of both the endocardium and the epicardium contour as indicated in the legend

put and keep the rest of the layers frozen during training. We can also keep them all or only keep a combination of them frozen. Since, the model learns low level features in the initial layers and high-level features in the last layers, retraining the deeper layers on the new dataset would intuitively result in better performances. In the context of using deep learning models with medical imaging datasets, retraining on all the layers has been seen to give better performance. (Kim et al., 2022) (Yu et al., 2019)

Using this approach, our first model was trained on Cine MRI dataset. Once we had a successful model, we used the weights from this model to initialize the training of our second model on the LGE-MRI dataset. Again, the final layer for both the Cine MRI and LGE MRI will be different depending on the size of the target vector for the boundary points representation of the myocardial contours.

3.3. Segmentation-Based Methods: 2D-Based Segmentation

In this section, we will discuss models that use segmentation-based techniques. The goal is to predict a segmentation mask from the input image by assigning a probability to each pixel in the image corresponding to the different classes to predict. In our case, the segmentation masks comprised left ventricular cavity and the myocardium. Since both 2D based methods using individual MRI slices and 3D-based methods, using 3D volumes were previously proposed to predict the segmentation masks, we will discuss each method in the following sections:

3.3.1. U-Net Architecture (Ronneberger et al., 2015)

This is the most standard architecture used for most of the segmentation tasks and has been extremely popular in medical imaging domain. This architecture utilizes a contracting and expanding path to assign a label to each pixel in our input image. The contracting path uses an encoder-based architecture which is basically the part of the same architecture that is described in Section 3.2.2 for its use in the regression-based method. This makes sense because in the contracting path, our goal is to understand how to recognize the region of interest inside the input image and learn the associated

features. For the localization part, we have the expanding path which serves as a decoder. The skip connections as indicated in the figure x are used to keep the spatial information intact in order to be able to place the detected labels in the original coordinates in the input image. In the end, we get a probability map for each pixel in the input image for which class label it might belong to. We use a softmax activation function to get the class labels with the highest probability for each pixel in the input image.

3.3.2. U-Net++ with Attention (Li et al., 2020)

In this slightly modified version of the standard U-Net, we used attention gates in the decoder phase of the network.

This architecture is the combination of U-Net with Attention (Oktay et al., 2018) and U-Net++ (Zhou et al., 2018). Both of them were designed to improve the retention of information through the skip connection paths. In U-Net++, dense convolutions are performed between the corresponding encoder and decoder through the skip connection path. Each dense block is fused with the up-sampled output of the lower dense block bringing the semantic level of the encoded feature closer to that of the feature maps waiting in the decoder. This makes optimisation easier when semantically similar feature maps are received.

This U-Net++ model was further modified by Li et al. (2020) to incorporate attention based U-Net (Oktay et al., 2018) into the U-Net++ architecture. This is also aimed at improving the performance in the skip connections of U-Net by applying attention gates. Attention gates help in capturing a sufficiently large receptive field in contrast with CNNs and thus, capture semantic contextual information. Attention gates being incorporated with the skip connections are depicted in Figure 8

3.3.3. TransUNet Architecture (Chen et al., 2021)

Transformer based architectures, while initially used for sequential data and Natural Language Processing (NLP), have also been recently introduced to image recognition and segmentation related problems (Dosovitskiy et al., 2021). One main advantage of a transformer-based architecture over a convolutional one is its receptive field. While the view of a convolutional neural network is limited to very local information, a transformer block basically sees the entire image. Intuitively, the transformer-based architecture makes up for a very good encoder. TransUNet (Chen et al., 2021) is a transformer-convolution hybrid architecture where the main contrast with the standard U-Net is that the bottleneck part of the encoder is replaced by a transformer block.
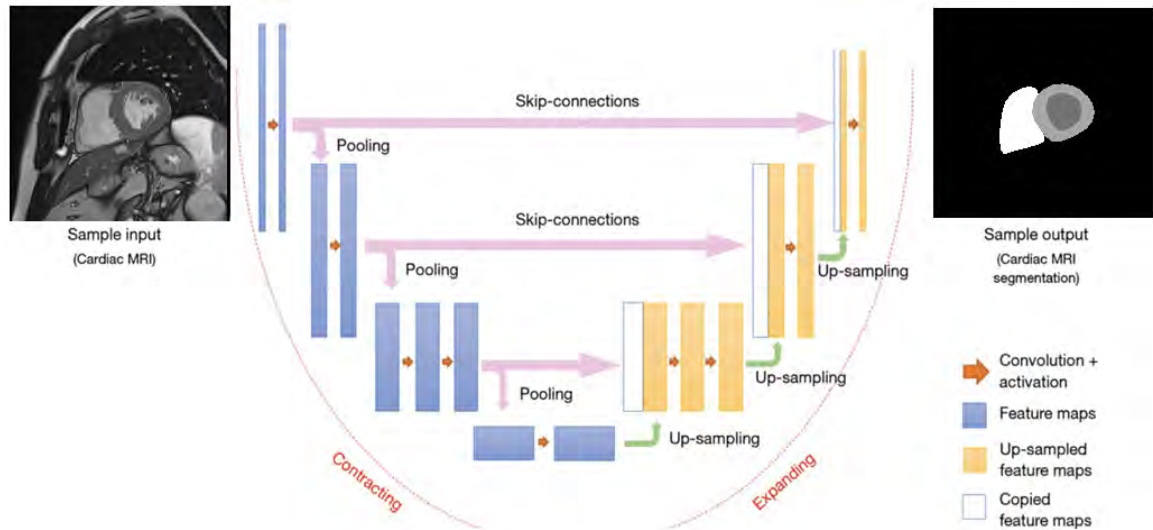
Figure 6: U-Net Architecture (Ronneberger et al., 2015)
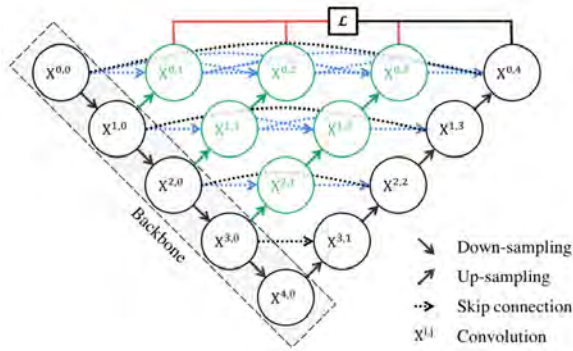
Figure 7: U-Net++ Architecture (Zhou et al., 2018)
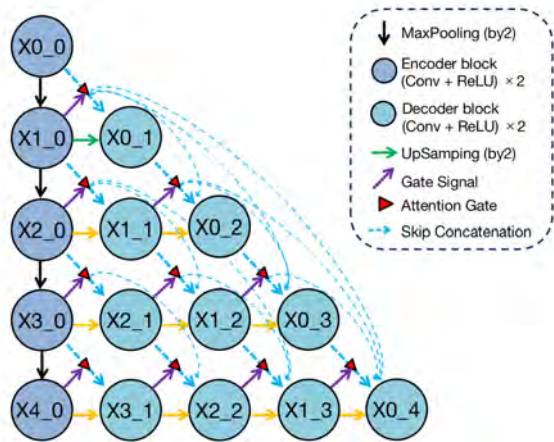
Figure 8: U-Net++ with Attention (Li et al., 2020)

3.4. Segmentation-Based Methods: 3D-Based Segmentation

In this method, we used the 3D volumes (contiguous slices for MRI) directly instead of using a slice by slice approach. This kept spatial information intact. Hence, this method is not adapted to part of our Cine MRI dataset which did not have any spatial depth.

3.4.1. UNETR Architecture (Hatamizadeh et al., 2022b)

UNETR is also a transformer-convolution hybrid architecture similar to the TransUNet. Its main difference with the TransUNet is that the whole encoder block has been converted from convolutional blocks to transformer blocks with the exception of a single layer. Again, the motivation here is to have more powerful encoder blocks. UNETR is a fairly new architecture but has seen some success in its use in the medical imaging domain (Rai et al., 2021) (Hatamizadeh et al., 2022a). Hence, we also wanted to observe its performance on Cardiac MRI segmentation.

3.5. Transfer Learning in Segmentation-Based Methods

Transfer Learning in Segmentation-Based Methods The transfer learning procedure here is very similar to what was discussed in Section 3.2.3 where transfer learning was discussed for regression-based method. In the figure x, we are looking at a simplified version of an encoder-decoder based architecture denoting the architectures we have encountered in this section. In the first stage, a model is trained on the Cine MRI dataset. After having a successful model, the weights from each of the layer are transferred to initialize the model that is to be used for segmenting the LGE MRI Dataset.
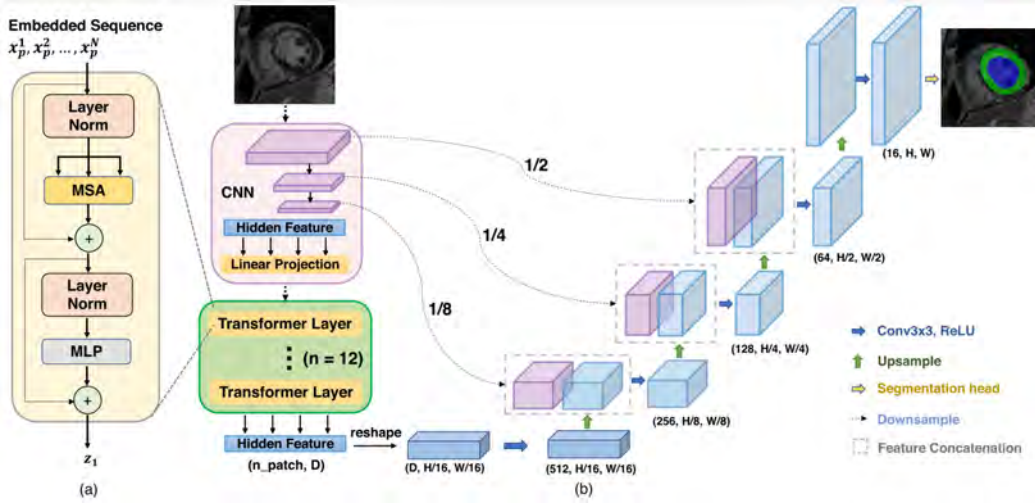
Figure 9: TransUNet architecture with transformer layer depicted where the encoder bottleneck block has been replaced (Chen et al., 2021)
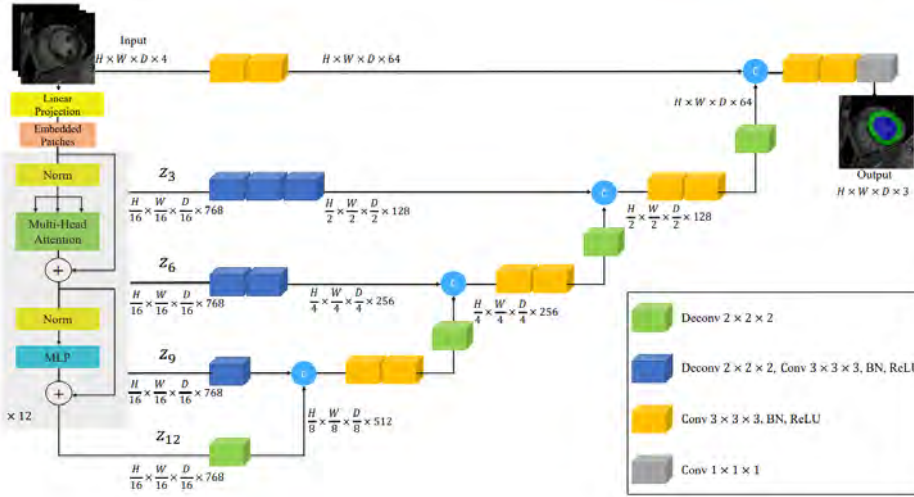
Figure 10: UNETR architecture with the transformer layers used in the encoder part indicated. (Hatamizadeh et al., 2022b)
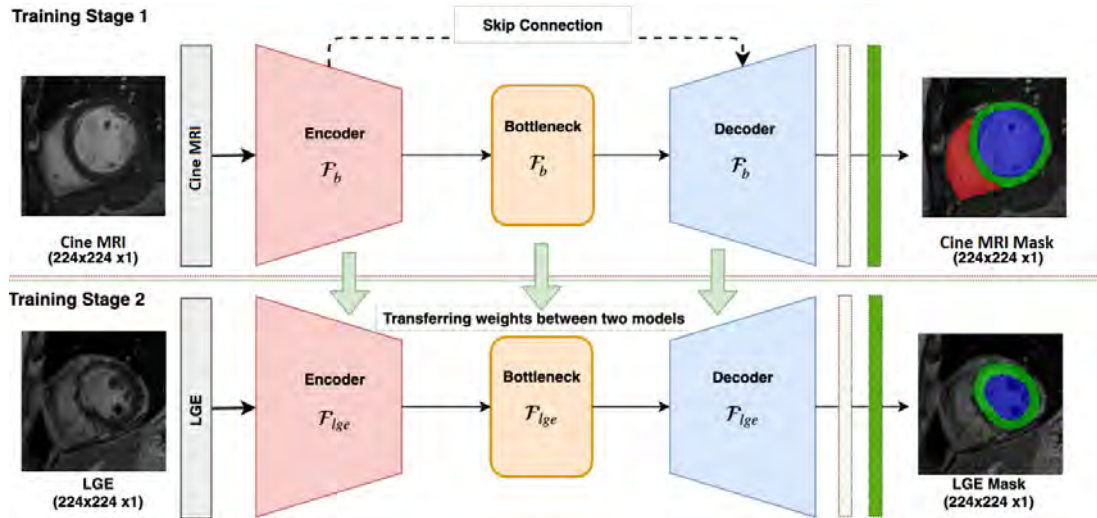
Figure 11: Transfer Learning as seen in U-Net based architectures for segmentation

3.6. Pre-processing

Our first step of pre-processing constitutes of focusing on the region of interest i.e the left ventricle and the myocardium of the human heart. Since the cardiac MRI has information besides this region of interest (as seen in Figure 12), the goal here is to move the region of interest to the center of the image and then to crop out or remove any excess part of the MRI outside of this region of interest.

The motivation for this is two-fold. The first case is empirical as we see the model being less vulnerable to the excess information besides the region of interest experimentally hence reducing the false positives. Secondly, from the study of the interpretability of the networks (Selvaraju et al., 2017)(Janik et al., 2021), we determine that it is indeed the ROI that the model needs to see and that the excess information does not contribute to the final predictions.

There are several methods to accomplish this, both segmentation as well as regression based (similar to what was described in Section 3.3.1 and Section 3.2.2, respectively). In the segmentation-based method, a deep learning based binary model is trained to detect the region of interest (left heart in our case) in the image. Using the binary mask of each input image, classical image processing is used to compute the centroid as well as a bounding box for the foreground i.e. ROI. The image is then translated to have the centroid at its center while the bounding box information is used to crop off the excess background. The size of the bounding box is empirically determined to be as small as possible without losing parts of the structure of interest. The regression-based methods are more straightforward in this case, a model is trained to simply predict the centroid and the main corners of the bounding box or even the anatomical landmarks of the heart. The same can be achieved using deep reinforcement learning. In our case, we used the segmentation-based method for this step.

Since the region of interest is not of the same size for all the patients/slices, once this step is complete, the input images are padded and center-cropped appropriately to have the uniform size of 224x224. The main motivation for using this specific resolution is to benefit from pre-trained weights in the encoder part of the used models.

Other pre-processing steps include the zero-one normalization which is implemented per image and volume for 2D and 3D-based methods, respectively. To tackle the contrast and noise related issue in the dataset, we also used contrast enhancement and additive Gaussian noise. This step is implemented as augmentative transforms to ensure varying level of contrast and noise adjustment for more robust results.

3.7. Training Configuration

All the experiments use Adam optimizer with a learning rate of 1e-04 and a batch size of 16.
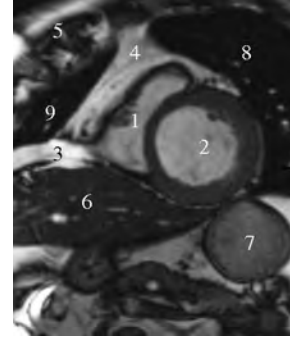
Figure 12: Labeled Cardiac MRI:- 1: Right Ventricle, 2: Left Ventricle, 3: Diaphragmatic Fat, 4: Paracardiac Fat, 5: Chest Wall, 6: Liver, 7: Stomach, 8: Left Lung, 9: Right Lung
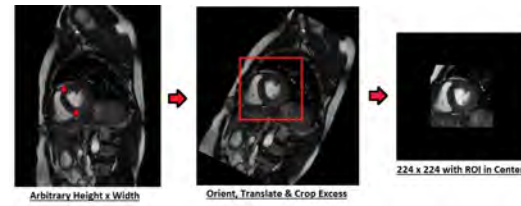
Figure 13: Pre-processing Steps

3.7.1. Segmentation-Based Methods

For the Cine MRI dataset, the number of epochs used are up to 30 with about one and a half hour of training time per epoch whereas for the LGE MRI dataset, the number of epochs used are up to 300 with about half a minute per epoch. For loss function, we tested Dice Loss, Generalized-Wasserstein Dice Loss (Fidon et al., 2017), Cross Entropy Loss and their combinations and results did not vary significantly. We thus used a customized version of the Dice Loss to account for missing labels as the standard dice loss function registered a complete loss (0% Dice Score) in cases of missing labels even where our model has successfully avoided the false positives. For the backbone architecture to be used in the encoder part of our segmentation based models, U-Net++ used VGG19 architecture with batch normalization. This was based on performance exploratory analysis and was seen in literature pertaining to medical imaging. (de la Rosa et al., 2021) (Jia et al., 2018)

3.7.2. Regression-Based Methods

For the Cine MRI dataset, the number of epochs used are up to 20 with about twenty minutes of training time per epoch whereas for the LGE MRI dataset, the number of epochs used are up to 500 with about twenty seconds of training time per epoch. For loss function, we used both L1(Mean Squared Error) and L2 (Mean Absolute Error) Loss depending on their performance for each particular case. The pre-trained deep CNN model used for the task was VGG19 architecture with batch normalization for the same reasons as mentioned in Section 3.7.1

3.8. Evaluation Metrics

The main evaluation metric being used is the Dice Score Coefficient(DSC) metric which is being supplemented by the Hausdorff Distance. The Dice score measures the global overlap of a predicted contour area (A_s) with the ground truth contour area (A_r). The value ranges between 0 and 1 (or 0 to 100%) with larger value indicating higher consistency between the predicted segmentation area and the ground truth.
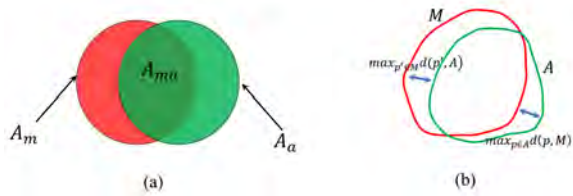
Figure 14: Graphic illustration of dice metric and Hausdorff distance. (a) Red (A_m) and green areas (A_a) are manually and automatically segmented, respectively. The intersection is represented by A_{ma}. (b) M and A denote manual and automated contour, respectively.

$$DSC(A_s, A_r) = 2A_{sr}/(A_s + A_r) \qquad (2)$$

On the other hand, the Hausdorff Distance (HD) measure the local agreement between to segmentation. Minimum distance is calculated between each point (p) on the ground truth mask or contour (R) to its nearest point (p') in the predicted mask or contour (S)

$$d(p, S) = min_{(p' \epsilon M)} \| p - p' \| \qquad (3)$$

Now, for all points in R and P, maximal distance $d(p, S)$ and $d(p', R)$ using (3) is calculated. Hausdorff distance is the maximum of these two distances. The value ranges between 0 and ∞, with smaller value indicating a higher consistency between the predicted segmentation area and the ground truth.

$$HD(A_S, A_R) = max(max_{(p \epsilon R)}d(p, S), max_{(p' \epsilon S)}d(p', R)) \qquad (4)$$

4. Results

As mentioned in Section 3.7, the results for regression-based methods were obtained using VGG19 architecture with batch normalization. For the sake of brevity, the results pertaining to 2D-Based segmentation come from U-Net++ with VGG19bn as backbone encoder as it gave better results than the TransUNet architecture. For quantitative results, see Table 4. More detailed quantitative analysis(see Figure 17) come from the model that performed the best. In this case, it is the 2D-based segmentation model used i.e U-Net++ with VGG19bn as the back bone encoder. In qualitative results, Figure 15 and 16 give example results from one basal, one middle and one apical slice. The good examples are cases where the model performs well on a challenging image from the test dataset. The difficult examples are challenging cases from the test dataset that the model struggles with.

5. Discussion

It was expected that, due to the relatively much smaller and lower quality of the LGE MRI dataset, the preliminary results before transfer learning would not be very good, in line with literature findings (Romero et al., 2019). However, as can be seen from Table 4, our model performed very well even before the use of transfer learning. One way to explain this would be that our model complexity allows us to capture what is needed from the dataset irrespective of its size. Another way to look at this would be that the size and variety (sites, quality, MI extent) of our LGE dataset i.e 1500 images was sufficient for a model to be trained with from scratch.

Transfer learning from the widely used cine SSFP towards LGE images induced a slight increase in performance independent of the used method. Such increase in performances came with a faster models' convergence. Since the LGE MRI dataset is acquired towards the end diastolic phase, we experimented transfer learning while focusing on the relevant time phases of Cine MRI, namely from diastasis to end-diastolic phase. This restriction to relevant time phrases, and accordingly to relevant heart shape had no significant effect regardless of the size of the data used. This suggested that the size of the source dataset used for training the first model in transfer learning might not always be relevant in terms of learning weights that the second model for the target dataset could benefit from.

Nonetheless, in general, we do have a successful segmentation of the left-ventricular cavity and the myocardium from LGE MRI images. As expected, segmentation performances were slightly lower in the apical slices as they are the most difficult to segment due to a very small amount of LV cavity present as well as to the higher presence of partial volume within the myocardial class. Tests of our model according to slice position, clinical centers where MRI exams were acquired and myocardial infarction reveals that our model has given quite robust results despite some outliers for which we believe the segmentation can be further improved using some post-processing techniques. From the qualitative results in Figure 15, we see that our model correctly predicted the segmentation masks despite the left-ventricular epicardium and endocardium boundaries being. The case from the apical slice in Figure 15c is a particularly difficult slice to segment. In figure 16, we see some challenging cases from the test dataset that the model has struggled with. These are slices mostly from the basal and apical positions.

In terms of model architecture from Table 4, at least for our case, we also observe that standard U-Net++

		(a) Cine MRI					
Transfer Learning	Method	Dice Score (%)			Hausdorff Distance (px)		
		Average	LV	Myocardium	Average	LV	Myocardium
n/a	Contour Regression	87.9 ± 3	91.5 ± 4	84.3 ± 4	2.54 ± 1.93	2.43 ± 1.84	2.66 ± 2.50
	2D Segmentation	93.2 ± 2	94.8 ± 4	91.6 ± 3	1.83 ± 1.68	1.72 ± 1.39	1.95 ± 2.04
	3D Segmentation	85.4 ± 4	90.5 ± 6	80.4 ± 5	2.99 ± 2.10	2.85 ± 1.59	3.14 ± 1.95
		(b) LGE MRI					
No	Contours Regression	86.6 ± 4	89.4 ± 5	83.7 ± 4	2.76 ± 2.10	2.54 ± 1.63	2.98 ± 2.31
	2D Segmentation	91.3 ± 3	93.1 ± 6	89.5 ± 4	2.51 ± 1.43	2.24 ± 1.18	2.78 ± 2.08
	3D Segmentation	85.1 ± 4	90.3 ± 4	80.0 ± 3	3.05 ± 1.83	2.89 ± 1.81	3.22 ± 2.13
Yes	Contours Regression	86.8 ± 4	89.7 ± 5	84.0 ± 4	2.63 ± 2.02	2.46 ± 1.71	2.81 ± 2.24
	2D Segmentation	91.6 ± 3	93.4 ± 6	90.0 ± 4	2.64 ± 1.28	2.28 ± 1.23	3.01 ± 1.62
	3D Segmentation	85.3 ± 4	90.6 ± 4	80.1 ± 3	3.01 ± 1.61	2.91 ± 1.75	3.12 ± 2.01

Table 4: Quantitative Results for (a) Cine MRI and (b) LGE MRI
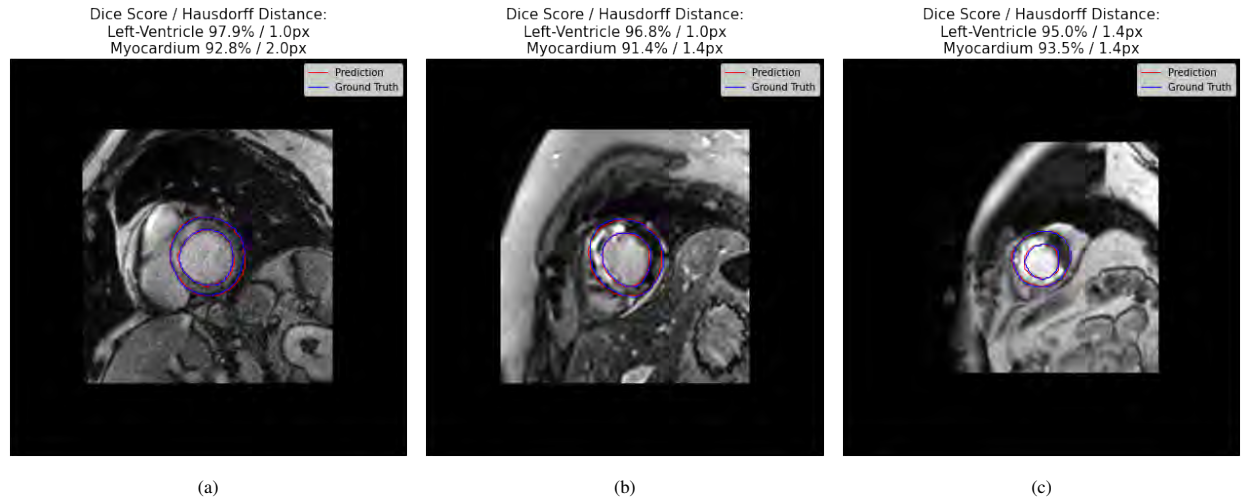
Figure 15: Example Good Case in (a) Basal, (b) Middle and (c) Apical Slice
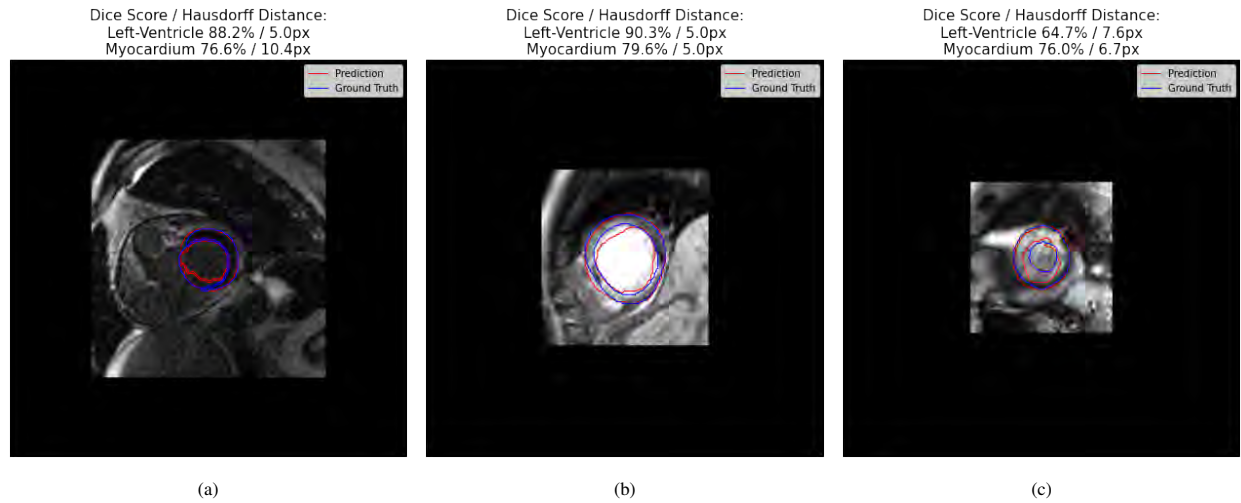
Figure 16: Example Difficult Case in (a) Basal, (b) Middle and (c) Apical Slice

9.12
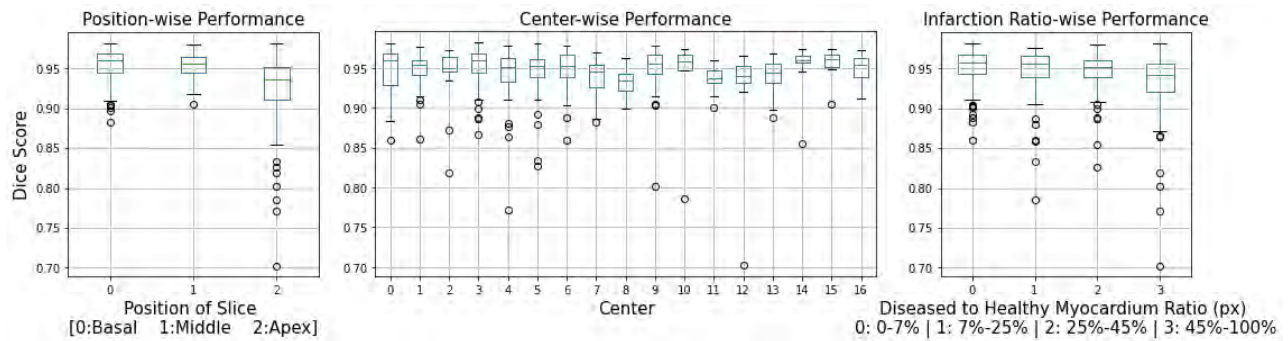
Figure 17: Box-Plots of the segmentation-based model performances through slice levels (left), imaging centers (middle) and infarct size (estimated as the scar to healthy myocardium ratio)

model with VGG19 encoder and batch normalization still outperforms transformer-based hybrid segmentation architectures. We also observe that although it had been an interesting and insightful approach regression-based methods are outperformed by the segmentation-based methods in the context of our task.

6. Conclusions

In this study, the significance of transfer learning approach from Cine MRI to LGE MRI was investigated. We performed regression and segmentation-based analysis and did a performance comparison between convolutional and transformer-based segmentation models. In conclusion, we perform well in segmenting left-ventricle and myocardium in the LGE MRI dataset but the contribution from transfer learning using Cine MRI was only minor in our data.

Acknowledgments

References

Baron, N., Kachenoura, N., Cluzel, P., Frouin, F., Herment, A., Grenier, P., Montalescot, G., Beygui, F., 2013. Comparison of various methods for quantitative evaluation of myocardial infarct volume from magnetic resonance delayed enhancement data. International Journal of Cardiology 167, 739–744. doi:10.1016/j.ijcard.2012.03.056.

Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P., Cetin, I., Lekadir, K., Camara, O., Ballester, M.Á.G., Sanroma, G., Napel, S., Petersen, S.E., Tziritas, G., Grinias, E., Khened, M., Varghese, A., Krishnamurthi, G., Rohé, M., Pennec, X., Sermesant, M., Isensee, F., Jaeger, P., Maier-Hein, K.H., Full, P.M., Wolf, I., Engelhardt, S., Baumgartner, C.F., Koch, L.M., Wolterink, J.M., Isgum, I., Jang, Y., Hong, Y., Patravali, J., Jain, S., Humbert,

O., Jodoin, P., 2018. Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved? IEEE Trans. Medical Imaging 37, 2514–2525. doi:10.1109/TMI.2018.2837502.

Brahim, K., Arega, T.W., Boucher, A., Bricq, S., Sakly, A., Mériaudeau, F., 2022. An improved 3d deep learning-based segmentation of left ventricular myocardial diseases from delayed-enhancement MRI with inclusion and classification prior information u-net (icpiu-net). Sensors 22, 2084. doi:10.3390/s22062084.

Brahim, K., Qayyum, A., Lalande, A., Boucher, A., Sakly, A., Mériaudeau, F., 2020. Efficient 3d deep learning for myocardial diseases segmentation, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 359–368. doi:10.1007/978-3-030-68107-4_37.

Camarasa, R., Faure, A., Crozier, T., Bos, D., de Bruijne, M., 2020. Uncertainty-based segmentation of myocardial infarction areas on cardiac MR images, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 385–391. doi:10.1007/978-3-030-68107-4_40.

Campello, V.M., Martín-Isla, C., Izquierdo, C., Petersen, S.E., Ballester, M.Á.G., Lekadir, K., 2019. Combining multi-sequence and synthetic images for improved segmentation of late gadolinium enhancement cardiac MRI .

Chen, C., Ouyang, C., Tarroni, G., Schlemper, J., Qiu, H., Bai, W., Rueckert, D., 2019a. Unsupervised multi-modal style transfer for cardiac MR segmentation, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 209–219. doi:10.1007/978-3-030-39074-7_22.

Chen, J., Li, H., Zhang, J., Menze, B.H., 2019b. Adversarial convolutional networks with weak domain-transfer for multi-sequence cardiac MR images segmentation, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 317–325. doi:10.1007/978-3-030-39074-7_34.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021. Transunet: Transformers make strong encoders for medical image segmentation .

Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J., 1995. Active shape models-their training and application. Comput. Vis. Image Underst. 61, 38–59. doi:10.1006/cviu.1995.1004.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA, IEEE Computer Society. pp. 248–255. doi:10.1109/CVPR.2009.5206848.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.

Du, X., Zhang, W., Zhang, H., Chen, J., Zhang, Y., Warrington, J., Brahm, G., Li, S., 2018. Deep regression segmentation for cardiac bi-ventricle MR images. IEEE Access 6, 3828–3838. doi:10.1109/ACCESS.2017.2789179.

Feng, X., Kramer, C.M., Salerno, M., Meyer, C.H., 2020. Automatic scar segmentation from DE-MRI using 2d dilated unet with rotation-based augmentation, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 400–405. doi:10.1007/978-3-030-68107-4_42.

Fidon, L., Li, W., García-Peraza-Herrera, L.C., Ekanayake, J., Kitchen, N., Ourselin, S., Vercauteren, T., 2017. Generalised wasserstein dice score for imbalanced multi-class segmentation using holistic convolutional networks .

Girum, K.B., Skandarani, Y., Hussain, R., Grayeli, A.B., Créhange, G., Lalande, A., 2020. Automatic myocardial infarction evaluation from delayed-enhancement cardiac MRI using deep convolutional networks, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 378–384. doi:10.1007/978-3-030-68107-4_39.

Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H., Xu, D., 2022a. Swin UNETR: swin transformers for semantic segmentation of brain tumors in MRI images .

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B.A., Roth, H.R., Xu, D., 2022b. UNETR: transformers for 3d medical image segmentation , 1748–1758doi:10.1109/WACV51458.2022.00181.

Hüllebrand, M., Ivantsits, M., Zhang, H., Kohlmann, P., Kuhnigk, J., Kühne, T., Schönberg, S.O., Hennemuth, A., 2020. Comparison of a hybrid mixture model and a CNN for the segmentation of myocardial pathologies in delayed enhancement MRI, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 319–327. doi:10.1007/978-3-030-68107-4_32.

Janik, A., Dodd, J., Ifrim, G., Sankaran, K., Curran, K.M., 2021. Interpretability of a deep learning model in the application of cardiac MRI segmentation with an ACDC challenge dataset .

Jia, H., Xia, Y., Song, Y., Cai, W., Fulham, M.J., Feng, D.D., 2018. Atlas registration and ensemble deep convolutional neural network-based prostate segmentation using magnetic resonance imaging. Neurocomputing 275, 1358–1369.

doi:10.1016/j.neucom.2017.09.084.

Kachenoura, N., Redheuil, A., Herment, A., Mousseaux, E., Frouin, F., 2008. Robust assessment of the transmural extent of myocardial infarction in late gadolinium-enhanced mri studies using appropriate angular and circumferential subdivision of the myocardium. Eur Radiol. doi:10.1007/s00330-008-0991-0.

Kim, H.E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M.E., Ganslandt, T., 2022. Transfer learning for medical image classification: a literature review. BMC Medical Imaging 22, 69. doi:10.1186/s12880-022-00793-7.

Lalande, A., Chen, Z., Pommier, T., Decourselle, T., Qayyum, A., Salomon, M., Ginhac, D., Skandarani, Y., Boucher, A., Brahim, K., de Bruijne, M., Camarasa, R., Correia, T.M., Feng, X., Girum, K.B., Hennemuth, A., Hüllebrand, M., Hussain, R., Ivantsits, M., Ma, J., Meyer, C., Sharma, R., Shi, J., Tsekos, N.V., Varela, M., Wang, X., Yang, S., Zhang, H., Zhang, Y., Zhou, Y., Zhuang, X., Couturier, R., Mériaudeau, F., 2021. Deep learning methods for automatic evaluation of delayed enhancement-mri. the results of the EMIDEC challenge. CoRR .

Lamy, J., Soulat, G., Evin, M., Huber, A., Cesare, A.D., Giron, A., Diebold, B., Redheuil, A., Mousseaux, É., Kachenoura, N., 2018. Scan-rescan reproducibility of ventricular and atrial MRI feature tracking strain. Comput. Biol. Medicine 92, 197–203. doi:10.1016/j.compbiomed.2017.11.015.

Li, C., Tan, Y., Chen, W., Luo, X., Gao, Y., Jia, X., Wang, Z., 2020. Attention unet++: A nested attention-aware u-net for liver CT image segmentation, in: IEEE International Conference on Image Processing, ICIP 2020, Abu Dhabi, United Arab Emirates, October 25-28, 2020, IEEE. pp. 345–349. doi:10.1109/ICIP40778.2020.9190761.

Liu, Y., Wang, W., Wang, K., Ye, C., Luo, G., 2019. An automatic cardiac segmentation framework based on multi-sequence MR image, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 220–227. doi:10.1007/978-3-030-39074-7_23.

Ly, B., Cochet, H., Sermesant, M., 2019. Style data augmentation for robust segmentation of multi-modality cardiac MRI, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 197–208. doi:10.1007/978-3-030-39074-7_21.

Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M.C.H., Heinrich, M.P., Misawa, K., Mori, K., McDonagh, S.G., Hammerla, N.Y., Kainz, B., Glocker, B., Rueckert, D., 2018. Attention u-net: Learning where to look for the pancreas .

Rai, H.M., Chatterjee, K., Dashkevich, S., 2021. Automatic and accurate abnormality detection from brain MR images using a novel hybrid unetresnext-50 deep CNN model. Biomed. Signal Process. Control. 66, 102477.

Romero, M., Interian, Y., Solberg, T.D., Valdes, G., 2019. Training deep learning models with small datasets .

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: Navab, N., Hornegger, J., III, W.M.W., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III, Springer. pp. 234–241. doi:10.1007/978-3-319-24574-4_28.

de la Rosa, E., Sidibé, D., Decourselle, T., Leclercq, T., Cochet, A., Lalande, A., 2021. Myocardial infarction quantification from late gadolinium enhancement MRI using top-hat transforms and neural networks. Algorithms 14, 249. doi:10.3390/a14080249.

9.14

Roth, H., Zhu, W., Yang, D., Xu, Z., Xu, D., 2019. Cardiac segmentation of LGE MRI with noisy labels, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 228–236. doi:10.1007/978-3-030-39074-7_24.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society. pp. 618–626. doi:10.1109/ICCV.2017.74.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition, in: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Vesal, S., Ravikumar, N., Maier, A.K., 2019. Automated multi-sequence cardiac MRI segmentation using supervised domain adaptation .

Wang, J., Huang, H., Chen, C., Ma, W., Huang, Y., Ding, X., 2019a. Multi-sequence cardiac MR segmentation with adversarial domain adaptation network, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 254–262. doi:10.1007/978-3-030-39074-7_27.

Wang, X., Yang, S., Tang, M., Wei, Y., Han, X., He, L., Zhang, J., 2019b. Sk-unet: An improved u-net model with selective kernel for the segmentation of multi-sequence cardiac MR, in: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A.A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges - 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers, Springer. pp. 246–253. doi:10.1007/978-3-030-39074-7_26.

Xue, W., Li, J., Hu, Z., Kerfoot, E., Clough, J.R., Öksüz, I., Xu, H., Grau, V., Guo, F., Ng, M., Li, X., Li, Q., Liu, L., Ma, J., Grinias, E., Tziritas, G., Yan, W., Atehortúa, A., Garreau, M., Jang, Y., Debus, A., Ferrante, E., Yang, G., Hua, T., Li, S., 2021. Left ventricle quantification challenge: A comprehensive comparison and evaluation of segmentation and regression for mid-ventricular short-axis cardiac MR data. IEEE J. Biomed. Health Informatics 25, 3541–3553. doi:10.1109/JBHI.2021.3064353.

Yang, S., Wang, X., 2020. A hybrid network for automatic myocardial infarction segmentation in delayed enhancement-mri, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 351–358. doi:10.1007/978-3-030-68107-4_36.

Yu, X., Zeng, N., Liu, S., Zhang, Y., 2019. Utilization of densenet201 for diagnosis of breast abnormality. Mach. Vis. Appl. 30, 1135–1144. doi:10.1007/s00138-019-01042-8.

Yue, Q., Luo, X., Ye, Q., Xu, L., Zhuang, X., 2019. Cardiac segmentation from LGE MRI using deep neural network incorporating shape and spatial priors, in: Shen, D., Liu, T., Peters, T.M., Staib, L.H., Essert, C., Zhou, S., Yap, P., Khan, A.R. (Eds.), Medical Image Computing and Computer Assisted Intervention - MICCAI 2019 - 22nd International Conference, Shenzhen, China, October 13-17, 2019, Proceedings, Part II, Springer. pp. 559–567. doi:10.1007/978-3-030-32245-8_62.

Zhang, Y., 2020. Cascaded convolutional neural network for automatic myocardial infarction segmentation from delayed-enhancement cardiac MRI, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 328–333. doi:10.1007/978-3-030-68107-4_33.

Zhou, Y., Zhang, K., Luo, X., Wang, S., Zhuang, X., 2020. Anatomy prior based u-net for pathology segmentation with attention, in: Puyol-Antón, E., Pop, M., Sermesant, M., Campello, V.M., Lalande, A., Lekadir, K., Suinesiaputra, A., Camara, O., Young, A.A. (Eds.), Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges - 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Springer. pp. 392–399. doi:10.1007/978-3-030-68107-4_41.

Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J., 2018. Unet++: A nested u-net architecture for medical image segmentation .

Zhuang, X., Xu, J., Luo, X., Chen, C., Ouyang, C., Rueckert, D., Campello, V.M., Lekadir, K., Vesal, S., Ravikumar, N., Liu, Y., Luo, G., Chen, J., Li, H., Ly, B., Sermesant, M., Roth, H., Zhu, W., Wang, J., Ding, X., Wang, X., Yang, S., Li, L., 2020. Cardiac segmentation on late gadolinium enhancement MRI: A benchmark study from multi-sequence cardiac MR segmentation challenge .
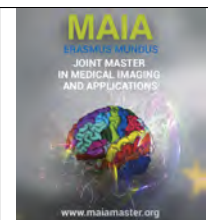
Evaluation of Automated Approaches for Lung Opacity Quantification

Raneim Nabil Hossni Mohamed, Adriyana Danudibroto

AGFA Radiology Solutions, Septestraat 27, 2640 Mortsel, Belgium

Abstract

At present, chest x-ray (CXR) is considered the primary tool for the detection and monitoring of lung abnormalities (Bradley et al., 2019). In the recent years, the main lung disease of interest is COVID-19, which is a new type of pneumonia caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) (Wang et al., 2020b). Monitoring lung opacity level is of great importance as it helps in determining the appropriate treatment and respiratory support for infected patients. The analysis of CXR by a trained radiologist is a time consuming and challenging task that involves inter-rater variability. Hence, AI can play an effective role in in automatic assessment and monitoring of lung severity using CXR. Brixia score (Borghesi and Maroldi, 2020) is a semi-quantitative multi-regional scoring system that proved a significant prognostic value in assessing lung severity from CXR in Italy during the pandemic. With the release of a large dataset of almost 5,000 CXR annotated with Brixia scores by only one experienced radiologist on the shift, automated solution can take part in reducing inter-rater variability and aiding radiologists especially during peak hours. The aim of this study is to explore the existing clinical and deep learning knowledge for developing automated solution to monitor lung involvement in pneumonia patients in general, including in COVID patients. Single-stage training and multi-stage training networks have been developed and experimented in this work using three datasets. The results seem promising in terms of consistency and robustness on the different datasets.

Keywords: Opacity, Brixia, Multi-stage, Single-stage, Pneumonia

1. Introduction

CXR is the most commonly performed radiological exam for evaluating the airways, pulmonary parenchyma and vessels, mediastinum, heart, pleura and chest wall (Pahiju and Thapa, 2017). It aids doctors in the diagnosis and monitoring of different lung conditions such as pneumonia, emphysema, and cancer. When X-ray passes through the body, different tissues absorb X-ray at different amounts. Air absorbs the least amount and appears black on CXR. When lungs are infected, the air filling the alveoli is replaced by a foreign substance (e.g. pus or blood) and appears white or opaque on CXR as shown in figure 1. These areas with increased densities are referred to as lung opacities (Lewis and Czum, 2013). Although, lung opacity does not indicate the pathological nature of lung abnormality, quantifying lung opacity from CXR can help in monitoring disease severity.

In December 2019, a novel viral pneumonia outbreak caused by the severe acute respiratory syndrome coronavirus2 (SARS-CoV-2) started in Wuhan, China (Zhu et al., 2020). With the fast spread of the disease, similar cases were reported in different parts of the world (Yasin and Gouda, 2020). High-resolution Computed Tomography (CT) was critical for investigating the novel coronavirus pneumonia especially in the early stage of the disease (Omar et al., 2020). However, the rising number of cases, the need to move infected patients around, high cost, and lack of experienced radiologists made it challenging to use CT especially in countries with limited resources. Consequently, CXR was the first-line triage substitute to aid in the diagnosis and prognosis (Harahwa et al., 2020). It's relatively cheap and mobile CXR can be easily brought to patient's bed including to the emergency departments. Although, CXR is not considered sensitive for the detection in early-stages (Jiang et al., 2020), it can be very useful in monitoring the rapid changes in lung abnormalities in COVID-19 patients, particularly in critical patients admitted to inten-

sive care units (Borghesi and Maroldi, 2020). The most
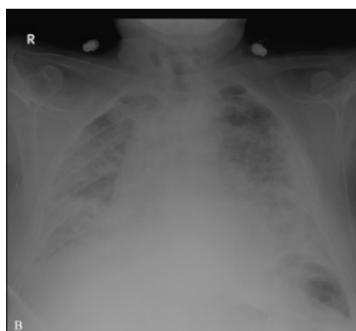
Figure 1: Bilateral lung opacities which appear more evident in the peripheral areas on CXR (Lomoro et al., 2020).

common radiologic findings in COVID-19 pneumonia are lung opacity changes (consolidations and/or ground-glass opacities), which are typically bilateral, peripheral, and located primarily in the lower fields (Chamorro et al., 2021). The wide range of possible disease manifestations from CXR and wide variability between radiologists in assessing lung involvement make lung opacity quantification very challenging (Litmanovich et al., 2020). For that reason, a precise quantification matrix of the severity and progression of lung aberrations is needed to determine the appropriate treatment and allocate hospital resources.

A new scoring system, Brixia (Borghesi and Maroldi, 2020), have been recently designed explicitly for COVID patients to map radiologists' findings to numerical values, leading to a more objective assessment and improved communication among specialists (Signoroni et al., 2021). According to Borghesi et al, Brixia scoring is very useful in ranking the stratification risk of COVID patients based on the severity of cases. However, the growing numbers of cases and limited number of experienced radiologists to assign Brixia score call for an automated solution for Brixia score prediction from CXR. In this research, various deep learning (DL) based approaches for predicting Brixia score were explored with the goal to quantify lung opacity in general.

2. State of the art

2.1. Scoring Systems

Given that radiographic findings are neither sensitive nor specific for COVID-19 detection as they overlap with other infections and pulmonary edema, CXR can be more valuable for assessing pulmonary infection severity (Li et al., 2020). Various semi-quantitative scoring systems have been proposed to reduce inter-rater variability among physicians in assessing the severity and progression of lung opacity. The most known and researched scoring systems are Brixia and Radiographic Assessment of Lung Edema (RALE) (Warren et al., 2018).

RALE score was developed to evaluate the degree and density of alveolar opacities on chest radiographs (Zimatore et al., 2021). It is determined by dividing the radiograph into four regions, defined vertically by the vertebral column and horizontally by the first branch of the left main bronchus, as shown in figure 2. Each quadrant is assigned a consolidation score from 0–4 based on the opacification percentage of the region and a density score from 1–3 (1=hazy, 2=moderate, 3=dense) based on the densities of the alveolar opacities. The final RALE score in the range [0,48] is calculated by summing the product of the consolidation and density scores for each quadrant (Warren et al., 2018). RALE was initially designed for lung edema, but has been adopted to quantify the severity of lung involvement in COVID pneumonia. It has been used in Queen Mary Hospital, Hong Kong; Pamela Youde Nethersole Eastern Hospital, Hong Kong; The University of Hong Kong; Shenzhen Hospital, Shenzhen; and University Hospital Careggi (Setiawati et al., 2021).
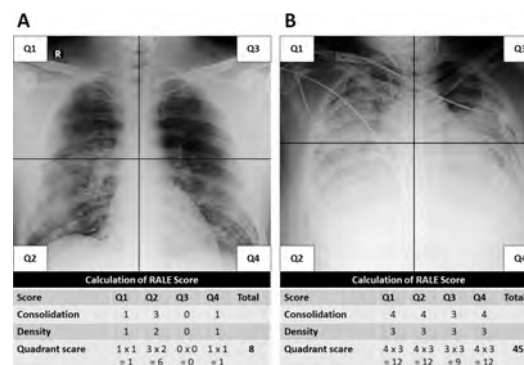
Figure 2: Example for RALE scoring system on two different cases (Homayounieh et al., 2020).

On the other hand, Brixia score was designed by the Radiology Unit 2 of ASST Spedali Civili di Brescia (Borghesi and Maroldi, 2020). It has been already implemented in routine reporting in Tongji Hospital, Wuhan and Azienda Socio Sanitaria Territoriale, Spedali Civili of Brescia, Italy (Setiawati et al., 2021). With this score, each lung is subdivided into three regions as shown in Figure 3. Each region is assigned a score:

- 0: no lung abnormalities
- 1: interstitial infiltrates
- 2: interstitial and alveolar infiltrates, interstitial dominant
- 3: interstitial and alveolar infiltrates, alveolar dominant

A final global score in the range [0,18] is calculated by summing the six scores.

Brixia scoring system has more detailed and complex indicators compared to RALE in scoring CXR to moni-

tor COVID-19 pneumonia. It has better localization capacity (Borghesi and Maroldi, 2020). The most important difference between the two scoring systems is that RALE can be assigned by a general practitioner due to its simplicity while Brixia has to be done by a trained radiologist (Setiawati et al., 2021). For that reason, automated solutions are helpful to aid inexperienced radiologists in assigning such scoring system. Automated solutions, in particular DL-based approaches, have high potentials to expand the role of chest imaging beyond diagnosis, to disease progression monitoring and risk stratification (Kundu et al., 2020).
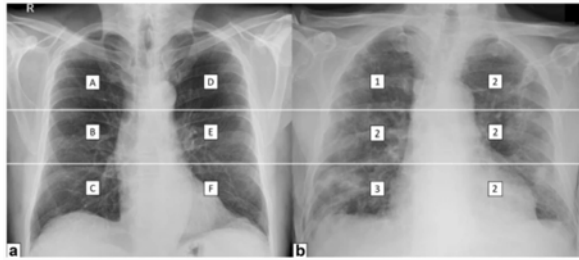
Figure 3: Lungs are first divided into six zones [A-F]. The upper line is drawn at the level of the inferior wall of the aortic arch. The bottom line is drawn at the level of the inferior wall of the right inferior pulmonary vein (Maroldi et al., 2021). An example of Brixia scoring system

From previous work on COVID classification (Tahir et al., 2022) and severity quantification (Signoroni et al., 2021), it is found that some deep learning architectures perform segmentation and alignment before classification. so in the following three subsections, different state-of-the-art methods in segmentation, alignment, and classification will be reviewed.

2.2. Segmentation

Over the past few years, DL demonstrated considerable capabilities in medical image segmentation and many algorithms have shown promising results on various segmentation tasks (Hesamian et al., 2019).

U-Net (Ronneberger et al., 2015) is the state-of-the-art in many biomedical image segmentation applications. It is formed by 2 main parts; an encoder to extract high-order abstract features from the data while reducing the spatial size of input images and a decoder that progressively recovers the original matrix size of the input image. Skip connections are used between the same levels on the encoder and decoder to retrieve fine details that were lost during the pooling operation (Çallı et al., 2021).

U-Net++ (Zhou et al., 2018) is a variant of U-Net in which skip connections are redesigned to enable feature aggregation at the varying-scale feature maps of the encoder and decoder sub-networks.

Another deep learning architecture that has achieved the state-of-the-art performance for medical imaging

object detection and semantic segmentation is the feature pyramid network (FPN) (Lin et al., 2017a). With its top-down architecture with lateral connections, it can extract high-level semantic feature maps at all scales(Li et al., 2021).

Although the previously mentioned networks give accurate results, they have relatively larger number of parameters compared to LinkNet. LinkNet (Chaurasia and Culurciello, 2017) was proposed to mitigate the problem of increased number of parameters and hence processing time. It decreases processing time by bypassing spatial information from encoder directly to decoder. LinkNet resembles the U-shape structure of U-Net but is different than U-Net in in the way that it replaces ordinary convolution structure with residual module and uses adding instead of stacking as a feature synthesis method.

On the other hand, Oktay et al. proposed a simple and effective solution, attention gates(AGs) modules, for segmenting ROI with various sizes and shapes with minimal computational overhead. AGs modules can improve the model sensitivity and prediction accuracy while preserving computational efficiency (Gaál et al., 2020). It can be easily inserted in Convolutional Neural Network (CNN) architectures such as U-Net where it implicitly learns how to highlight relevant task features and suppress irrelevant regions (Oktay et al., 2018).

2.3. Alignment

The most widely used DL-based registration methods are an encoder–decoder CNN, a Spatial Transformer Network (STN) (Jaderberg et al., 2015), and a Generative Adversarial Network (GAN) (Chen et al., 2021). An encoder-decoder network, U-Net, can be used for image registration by taking the moving and fixed images as inputS and predicting the deformation field.
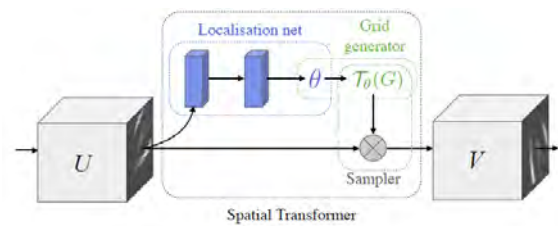
Figure 4: The architecture of a spatial transformer module (Jaderberg et al., 2015).

On the other hand, STN is a differentiable module that can be inserted anywhere in the network, giving the network the ability to spatially transform an image or feature map without extra training supervision (Jaderberg et al., 2015). It has been used in most DL image registration methods, especially unsupervised/weakly-supervised methods. As shown in figure 4, STN consists of three parts: a localisation network, a grid generator and a sampler. The localization network is a

simple CNN that learns the transformation parameters. With the output from the localization net, transformation parameters, grid generator generates a sampling grid, T(G), which is applied to the input by the bilinear sampler to produce the warped output (Jaderberg et al., 2015).

GAN-based image registration combines a U-Net and an STN as a generator to warp the moving image. The discriminator differentiate between the warped moving image and the fixed image to aid the generator in predicting a high-similarity warped moving image to the fixed image.

2.4. Classification

Medical image classification has been improved and accelerated by the advent of Transfer Learning (TL). TL can improve the performance on a new task by leveraging the knowledge learned in advance of similar tasks. Given that, TL is effective in overcoming data scarcity. Pretrained VGG-16 (Simonyan and Zisserman, 2014), ResNet50 (He et al., 2016), Inceptionv3 (Szegedy et al., 2016), EfficientNet (Tan and Le, 2019) are widely used in research and industry for image classification.

Another CNN architecture that has shown the-state-of-the-art is RetinaNet (Lin et al., 2017b). It was intially designed for object detection task, but has been adopted as a baseline for classification problems. The architecture of RetinaNet consists of ResNet34-Feature Pyramid Network and two subnetworks for classification and bounding box regression.

Vision Transformer (ViT) has demonstrated excellent results compared to state-of-the-art CNN while requiring substantially fewer computational resources to train (Dosovitskiy et al., 2020). In ViT, the image is interpreted as a sequence of patches of fixed size. A lower-dimensional linear embedding is created from the vectorized patches using trainable linear layer. To retain the positional information of the patches, position embeddings are added to patch embeddings. The resulting sequence of vectors is fed to the ViT encoder.

2.5. Severity Assessment

Short after the pandemic breakout, many researchers shifted their focus to chest imaging in managing COVID. Most of the proposed systems aimed to detect COVID or classify it against other lung diseases in either CXR or CTs. Table 1 summarizes the DL approaches that will be discussed in this section in details.

One research group designed a deep convolutional neural network, COVID-NET, to classify between normal, COVID infection, and non COVID infection leveraging residual architecture design principles. COVID-Net network architecture is made up of lightweight residual projection-expansion-projection-extension (PEPX) (Wang et al., 2020a) design pattern which consists of:

- 1×1 convolutions for projecting input features to a lower dimension
- 1×1 convolutions for expanding features to a higher dimension
- Efficient 3×3 depth-wise convolutions for learning spatial characteristics to minimize computational complexity while preserving representational capacity
- 1×1 convolutions for projecting features back to a lower dimension
- 1×1 convolutions that finally extend channel dimensionality to a higher dimension to produce the final features

It was the first open source code of a network designed for COVID-19 detection from CXR images and first release of a large public dataset containing 13,800 chest X-ray images on 13,645 patients. Their network reached accuracy of 92.4% for COVID classification. This study paved the way for other researchers to propose other networks for COVID detection in CXR. Some other studies utilized pretrained networks such as ResNet50, InceptionV3, and VGG16, and fine-tuned them on COVID datasets or used ensambling of multiple modified versions of them after fine-tuning (Gour and Jain, 2022) (Pham, 2020) (Kumar et al., 2022).

Another group built on COVID-NET and named their network COVID-NET S in which they replaced the last layers of COVID-Net with a set of dense layers (a 128 neuron dense layer, a 3 neuron dense layer, and a single output score prediction layer) (Wong et al., 2021). Data consisted of 396 CXRs that are annotated by two board-certifed expert chest radiologists (with 20+ years of experience) and a 2nd-year radiology resident. The scoring system consisted of geographic extent and opacity extent adapted from (Wong et al.10) and (Warren et al.11) for each lung.

Geographic Extent:

- 0: no involvement
- 1: 25% involvement
- 2: 25–50% involvement
- 3: 50–75% involvement
- 4: 75% lung involvement

Opacity extent:

- 0: no opacity
- 1: ground glass opacity
- 2: mix of consolidation and ground glass opacity (less than 50% consolidation)
- 3: mix of consolidation and ground glass opacity (more than 50% consolidation)
- 4: complete white-out

100 versions of the network were independently trained (50 to predict geographic extent scoring and 50 to predict opacity extent scoring) using random subsets of CXRs from the study and the network was evaluated using stratifed Monte Carlo cross-validation experiments. The network achieved R_2 of 0.739 and 0.741 between predicted scores and radiologist scores for geographic extent and opacity extent respectively.

Another study leveraged the use of pretrained Densenet models on 7 non-Covid public datasets to extract general representations about lungs and other aspects of CXRs (Cohen et al., 2020). Then, they used linear regression to predict the severity scores from 96 Covid CXR. The severity scores were performed by three blinded experts: two chest radiologists (with 20+ years of experience) and a radiology resident on 96 CXR. The scoring system was similar to (Wong et al., 2021) as it combined extent of lung involvement with ground glass opacity or consolidation and opacity extent score for each lung. However, their opacity extent score ranged from 0-3 instead of 0-4 (0 = no opacity; 1 = ground glass opacity; 2 = consolidation; 3 = white-out).

Another research group introduced the idea of using a pretrained convolutional Siamese neural network–based algorithm. In Siamese neural network, two input images are passed through identical subnetworks with shared weights and then a euclidean distance is calculated between the final two layers of the network. In that way, one image of interest can be compared to a pool of healthy CXR and the disease severity can be estimated as the median of those Euclidean distances, named PXS score (Li et al., 2020). The model was pretrained on approximately 160,000 anterior-posterior images from CheXpert and transfer learned on 314 COVID-19 frontal chest radiographs. The algorithm was evaluated on 167 radiographs and PXS scores were correlated with modified RALE assigned by two thoracic radiologists and one in-training radiologist. The PXS score and the direction of change in PXS score in follow-up agreed with the assigned modified RALE score.

The most related previous work to our research is from (Signoroni et al., 2021). In this paper, they introduced an end-end deep learning pipeline named BS-Net designed to handle different tasks (segmentation, spatial alignment, and score estimation) and trained "from-the-part-to-the-whole" on different datasets (three public datasets for segmentation, synthetic dataset for alignment, and their own collected Brixia dataset). They used U-Net++ (Zhou et al., 2018) for segmentation, Spatial transformer (Jaderberg et al., 2015) for alignment, and Retina classifier for Brixia scores estimation. Their system predictions outperformed single human annotators in terms of accuracy and consistency.

From the literature review, it is noticed that the current research is focused on the binary classification and detection of lung diseases more than quantifying severity of the disease. Hence, in this work, we are quantifying lung opacity using Brixia scoring system. Although some recent studies have investigated how AI can aid radiologists in lung opacity quantification, they are limited in their scope due to the lack of multi-reader datasets and the absence of ablation studies or comparisons of their models. To address these limitations, in this study, several state of art techniques have been tested and harmonized to form automated solution for Brixia score predictions. We propose a single-stage and multi-stage networks and compare their performance on different datasets to understand the effect of segmentation and alignment on the classifier performance in predicting Brixia scores. In addition, we study the effect of using ViT vs CNN classifiers for producing consistent predictions of Brixia scores on different datasets annotated by multiple radiologists.

3. Datasets

3.1. Segmentation Datasets

Three public datasets were combined and used for training the segmentation module. Montgomery County (Jaeger et al., 2014) dataset consists of 128 X-ray images with 80 healthy lungs and 58 diseased ones by tuberculosis. Data has been acquired by the Department of Health and Human Services, Montgomery County in Maryland, USA. JSRT databases (Shiraishi et al., 2000) was released by Japanese Society of Radiological Technology (JSRT). It contains 247 images with 154 cases of lung nodules and 93 healthy cases. Shenzhen Hospital (Gordienko et al., 2018) was acquired from Shenzhen No. 3 People's Hospital in Shenzhen, China. It is also a tuberculosis X-ray images; however, lung masks are only available for 566 cases which will be used in this study. X-ray images in the three combined set were resized to 512x512 and standardized using min-max normalization. Dataset was divided into 223 (first 50 of Montgomery County and Shenzhen Hospital and original split of JSRT (123 test)) images for test and 728 images for training.

3.2. Alignment Dataset

The same images used for segmentation were used for generating the alignment dataset. As shown in table 2, different image transformations were applied to masks of the segmentation dataset using albumentations library to generate synthetic alignment dataset.

3.3. Brixia Dataset

The Brixia dataset includes 4,707 CXR images of COVID-19 subjects for both triage and patient monitoring in sub-intensive and intensive care units. It is collected between March 4th and April 4th 2020 of pandemic peak at the ASST Spedali Civili di Brescia. All

Method	Dataset	Performance	Task
COVID-Net	13,975 CXRs	Accuracy: 93.3%	COVID classification
Modified COVID-Net (Transfer Learning)	396 CXRs 13,975 CXRs	R_2: 0.739 and 0.741 between predicted scores and radiologist scores for geographic extent and opacity extent	Severity Assesment
Siames Neural Network	160,000 CheXpert 314 COVID patient set	R: 0.86 between PXS score and RALE	Severity Assesment
BS-Net	4,703 CXRs Brixia 192 CXRs Chohen	MAE: 0.471 Brixia MAE: 0.490 Cohen	Severity Assesment

Table 1: Summary of existing Severity Assesment and COVID classification work
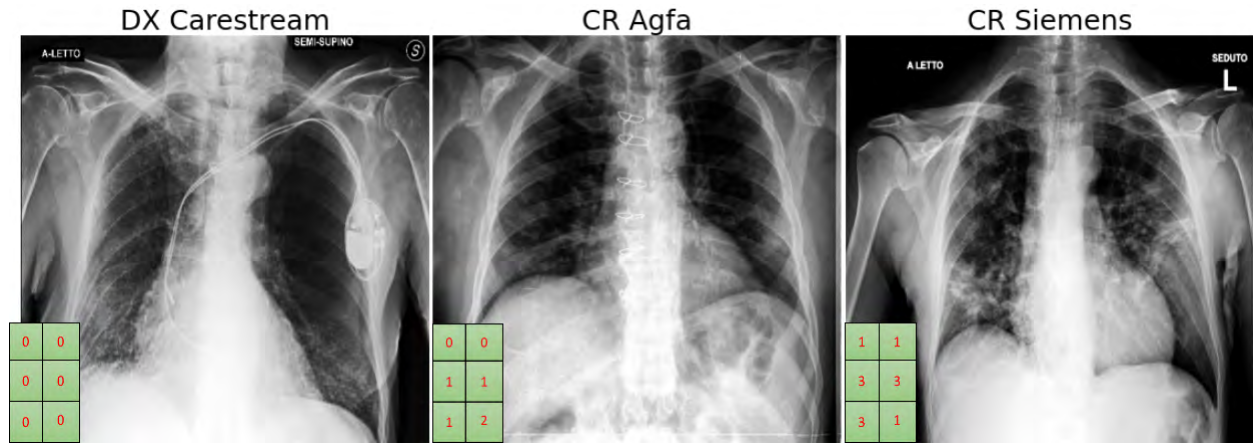
Figure 5: Samples from the Brixia dataset from different manufacturers with their corresponding Brixia scores.

Transformation Operation	Parameters	Probability
Rotation	25 degree	0.8
Scale	10%	0.8
Shift	10%	0.8
Elastic transformation	alpha=60, sigma=12	0.2
Grid distortion	step=5, limit=0.3	0.2
Optical distortion	distort=0.2, shift=0.05	0.2

Table 2: Synthestic dataset transformation parameters (Signoroni et al., 2021)

Parameter	Value
Modality	CR (62%) - DX (38%)
View Position	AP (87%) - PA (13%)
Manufacturers	Carestream, Siemens , Agfa

Table 3: Brixia dataset details (Signoroni et al., 2021)

images in Brixia dataset are annotated by the radiologist on the shift who is part of about 50 radiologists in the hospital with extensive years of experience. Each image has six scores in the format of a string of six digits. The global score can be calculated by simply summing the six scores. The age and sex of subjects is provided as well. The dataset is anonymized and approved by the local Ethical Committee (NP4121) for research purposes usage. Three images from three different manufacturers with their corresponding labels are shown in figure 5. Details about manufacturers, modalities and view position are shown in table 3.

Brixia dataset comes in dicom format, so images were first imported from the DICOM files. Preprocessing of the data followed the original paper preprocess-

ing (Signoroni et al., 2021). Image pixel values were mapped between 0 and 1. Then, data normalization was achieved by applying an adaptive histogram equalization (CLAHE, clip:0.01) to adjust image contrast, a median filtering to mitigate noise (kernel size: 3), and a clipping outside the 2nd and 98th percentile to drop the outliers.
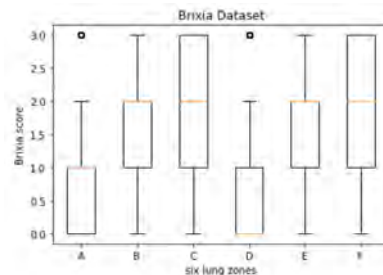
Figure 6: Score distribution of the radiologist on the shift annotations for Brixia dataset

3.4. Cohen Dataset

To test the final model robustness and generalization capabilities, a well known COVID public dataset (Cohen et al. (2020b) was used. It is collected in different centers all over the world. The Brixia score of this dataset was provided by two board certified radiologists with 22 and 2 years of experience. In the process of labeling, few hard cases were discarded due to too low resolution or significant mispositioning. The final annotated dataset contains 192 CXRs of positive or sus-

pected COVID patients. Figure(7) shows the distribution of the senior annotations while figure(8) shows the distribution of the junior radiologist respectively.

3.5. Consensus Dataset

This subset is publicly available with the Brixia dataset. To make this set, four different radiologists were asked to annotate a subset of 15 CXRs from the Brixia dataset to assess the inter-rater variability. Three of the chosen radiologist are with 9, 15 and 22 years of experience and one is at 2nd year of training. The mode of the four radiologists with the radiologist on shift annotations was provided as the new label for this subset and the distribution of the final annotations is shown in figure 9.
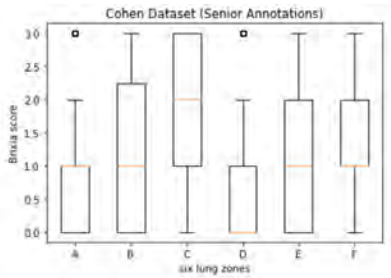
Figure 7: Score distribution of the senior radiologist annotations for Cohen dataset
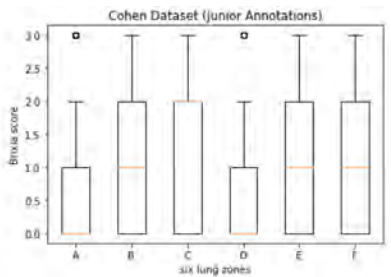
Figure 8: Score distribution of the junior radiologist annotations for Cohen dataset
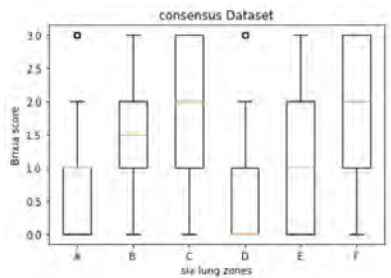
Figure 9: Score distribution of the mode of five radiologists annotations for consensus dataset

4. Material and methods

4.1. Data Augmentation

Data augmentation helps segmentation and classification models to generalize results to unseen data as it adds variations in limited training data and thus helps to avoid overfitting. In addition, it is very applicable in lung segmentation and classification tasks since it can help in handling CXR variability resulted from image acquisition settings such as patient positioning and dose variation. In the segmentation part, different combinations of random rotations, shifts, blur, brightness, and contrast were applied using the fast Albumentations library where the probability of applied transformation can be specified using the P parameter. Table (2) shows the different ablumentations tested. For the classification part, same transformations were tested without the horizontal flip because the labels will need to be changed as well with respect to the flip. However, rotation with 25 degrees was tested because it can resemble incorrect positioning of patients in real life and also does not require changing the ground truth labels.

4.2. Segmentation

Detecting the region of interest (ROI), that is the lungs, is crucial to have accurate diagnosis and prognosis of the diseases. The main challenge is that lung fields are opacified due to pneumonia. These opacities frequently alter the intensity values of the lungs, so lung masks can be incorrectly predicted by segmentation models leading to inaccurate lung segmentation (Souza et al., 2019). Taking this into consideration, the segmentation training datasets were chosen to have lung opacities, so that segmentation models can learn how to segment the opacified lungs from the target Brixia dataset used in inference.

Several state-of-the-art segmentation models (U-Net, U-Net++, FPN, LinkNet, Attention U-Net, and Attention ResU-Net) have been evaluated to choose the best performing one as the first module for the final end-to-end network. To have a fair comparison among these models, preprocessing, augmentation, and hyperparameters were maintained throughout the segmentation experiments. Models were trained for 50 epochs with batch size 8 and adam optimizer. The losses used for optimize segmentation models are the dice loss and the binary focal loss. This combination mitigates the class imbalance between foreground and background and the easy and hard to classify examples. For quantitative assessment of the different segmentation models, Dice similarity coefficient metric (DSC) and intersection over union (IoU) metric were used on the segmentation dataset since they have ground truth masks.

$$DSC = \frac{2|GT| \cap |S|}{|GT| + |S|} \qquad (1)$$

$$IoU = \frac{|GT| \cap |S|}{|GT| + |S|} \quad (2)$$

On the other hand, different models were only qualitatively analyzed on the target dataset, Brixia, since it does not have masks.

4.3. Alignment

Image alignment is a crucial step to establish optimal correspondence within images taken at different times across different patients and to enable direct comparisons between multiple scans of the same patient (Gaál et al., 2020). Since Brixia scores are classified per lung region, lung alignment is even more important to locate the correct lung field in the six regions of interest. Since we did not have a fixed reference image, STN was the optimal solution for alignment in our project. STN was experimented twice; as a differentiable module and as a separate model. In one experiment, it was inserted between convolution layers where it was applied on the backbone feature maps and trained with the classification part. In another experiment, a pretrained STN (Signoroni et al., 2021) on the synthetic dataset described in section 3 was combined with segmentation and classification as a separate pretrained block. To train the STN as a separate model, the synthetic masks (transformed) are used as the images and original masks are used as the ground truth masks. Then, STN is trained on dice loss to learn how to align the transformed masks to the original masks.

4.4. Brixia Classification

For the final scoring module, ResNet, FPN, and ViT were investigated. ResNet was tested alone as a baseline classifier. Then, it was combined with alignment and segmentation in which the same encoder used for segmentation was used for classification with weights transferred from segmentation task and fine-tuned by Brixia dataset. The second classifier experimented was inspired by RetinaNet architecture. The multi-scale feature maps from the segmentation backbone were enhanced with top-down pathway and lateral connections of FPN and then convolution layers or fully connected layers were attached to different FPN levels for Brixia score predictions. A more sophisticated classifier, ViT, has been tested since it has shown better performance for COVID-19 detection over CNNs (Park et al., 2021). However, it requires huge amount of data if trained from scratch. Instead, hybrid ViT has shown better performance in small-sized data set. Hybrid ViT (Park et al., 2021) utilizes a CNN backbone that extracts initial low-level feature embedding which is used later for training the transformer. So, the backbone used in segmentation was used to extract the intial features that was used after to train ViT-B/16. The classification models were compared with respect to the mean absolute error (MAE). Classification models were optimized using the cross entropy loss function and were trained for 80 epochs with 8 batch size and adam optimizer.

4.5. Merging: Final Architectures

After experimenting each block, three different architectures were developed and compared against each other.

The FPN-based multi-stage network is shown in figure 10. The preprocessed input image passes through ResNet (He et al., 2016) backbone, a series of convolution blocks that extracts feature maps at different scales. Since the aim is to have end-to-end framework, the same ResNet18 backbone is used for the classification branch with weights transferred from the segmentation task. For segmenting the lungs, FPN with ResNet18 backbone is used as it produces feature maps that can be both semantically and spatially strong. The output from the segmentation block is used as an input to the alignment block to estimate the transformation parameters that is used after by the resampler to align the feature maps of ResNet18 backbone and align the mask. The aligned mask is then multiplied by the aligned features to give attention to the lungs. Then, ROI pooling module is applied on the aligned feature maps to extract six lung regions with a vertical overlap of 25%. This pooling module provides the network with prior information about the location of the six Brixia score regions. The output from ROI is passed after to the classifier to predict the Brixia score. The classifier block utilizes another FPN where the higher level, semantically stronger feature map is exploited for the final predictions. The shape of the output predictions is 3x2x4 where 3x2 is the six regions of the lung and 4 is number of classes of Brixia score.

In ViT multi-stage architecture, ViT was placed between the FPN and dense layers and segmentation and alignment were the same exact ones used in the previous architecture. The output from ViT passes through a dense layer to produce the final predictions.

The ViT single-stage network is shown in figure12. The architecture consists of a ResNet backbone, ViT, ROI pooling, and a dense layer. The main difference between this architecture and the previous two architectures is that this network is trained at once and does not have a segmentation block or alignment blocks.

5. Results

In this section, we first evaluate the performance of the segmentation block and visualize the results on Brixia dataset. Then, we study the effect of adding segmentation, pretrained STN, and FPN blocks to the baseline ResNet18 backbone to form the final FPN_based network. After that, we show the difference in performance when using pretrained STN, STN trained with
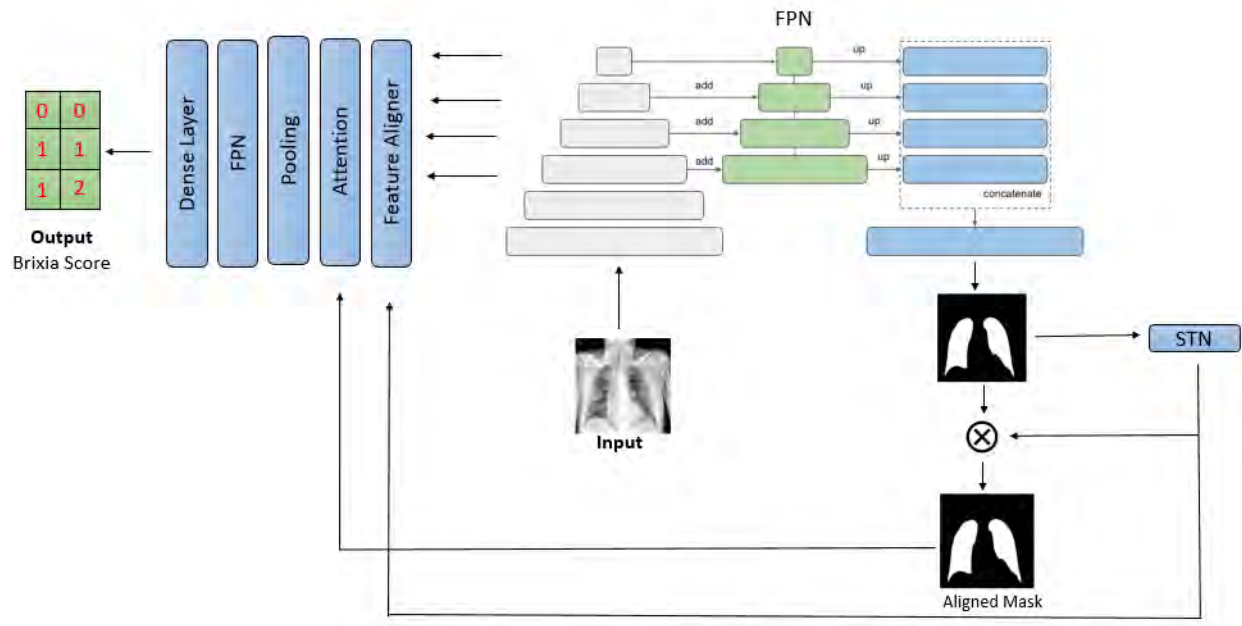
Figure 10: FPN-based multi-stage model
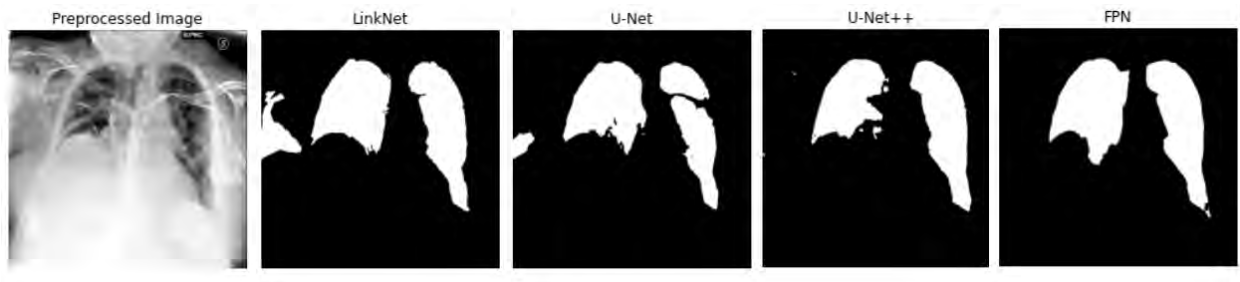
Figure 11: Segmentation masks from the four highest performing models on an image from Brixia dataset

Model	Backbone	IoU	DSC	# Parameters
FPN	VGG16	0.940	0.967	19.315M
	EfficientNetB0	0.943	0.971	8.790M
	ResNet18	0.945	0.972	15.569M
U-Net	VGG16	0.940	0.969	19.037 M
	EfficientNetB0	0.946	0.972	10.114M
	ResNet18	0.943	0.970	14.334M
U-Net++	VGG16	0.935	0.966	26.147M
	EfficientNetB0	0.940	0.968	14.274M
	ResNet18	0.942	0.970	18.267M
LinkNet	VGG16	0.944	0.971	15.603M
	EfficientNetB0	0.946	0.972	6.095M
	ResNet18	0.943	0.971	11.515M

Table 4: Performance of different segmentation models in terms of DSC, IoU, and number of parameters

10.9

	ResNet18	ResNet18+Segm	ResNet18+Segm+Allign	ResNet18+Segm+Allign+FPN
			Brixia Dataset	
A	0.571±0.723	0.557±0.685	0.507±0.642	0.527±0.628
B	0.620±0.650	0.657±0.678	0.593±0.645	0.571±0.628
C	0.723±0.709	0.716±0.726	0.655±0.697	0.591±0.639
D	0.525±0.732	0.525±0.720	0.484±0.668	0.448±0.626
E	0.746±0.734	0.755±0.796	0.665±0.722	0.661±0.687
F	0.712±0.716	0.842±0.795	0.7186±0.743	0.725±0.709
Global	2.431±1.987	2.426±0.795	2.309±1.962	1.983±1.704

Table 5: Brixia score predictions performance in terms of MAE and STD when adding blocks to the baseline ResNet18 model to reach the final FPN-based model(ResNet18+Segm+Allign+FPN)
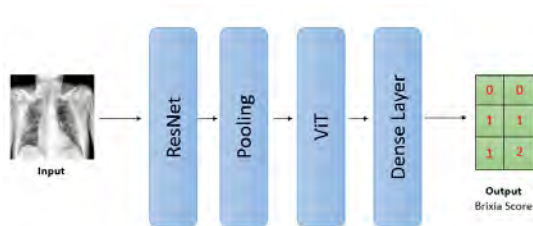
Figure 12: ViT single-stage model

	Brixia Dataset	Consensus Dataset
	MAE	
A	0.431±0.618	0.46±0.639
B	0.550±0.653	0.547±0.606
C	0.518±0.582	0.540±0.607
D	0.450±0.647	0.440±0.627
E	0.533±0.631	0.527±0.574
F	0.614±0.655	0.587±0.613
Global	1.945±1.647	2.087±1.68

Table 8: Perfomance of ViT multi-stage stage model on Consensus and Brixia datasets in terms of MAE and STD

	STN layer	STN block	Without STN
		MAE	
A	0.490±0.618	0.433±0.59	0.4947±0.648
B	0.701±0.733	0.544±0.640	0.557±0.662
C	0.689±0.725	0.529±0.614	0.601±0.654
D	0.429±0.631	0.399±0.584	0.439±0.632
E	0.731±0.781	0.586±0.703	0.597±0.689
F	0.755±0.711	0.652±0.692	0.655±0.697
Global	2.665±2.223	2.02±1.73	2.124±1.853

Table 6: Brixia score prediction performance in MAE and STD when using pretrained STN,when training STN with classification, and when removing STN from the architecture

	Brixia Dataset	Consensus Dataset
	MAE	
A	0.527±0.628	0.340±0.540
B	0.571±0.628	0.413±0.624
C	0.591±0.639	0.393±0.576
D	0.448±0.626	0.293±0.560
E	0.661±0.687	0.493±0.651
F	0.725±0.709	0.440±0.616
Global	1.983±1.704	1.520±1.753

Table 9: Perfomance of FPN-based multi-stage model on Consensus and Brixia datasets in terms of MAE and STD

	Brixia Dataset	Consensus Dataset
	MAE	
A	0.463±0.617	0.293±0.497
B	0.576±0.644	0.447±0.606
C	0.478±0.586	0.420±0.545
D	0.422±0.596	0.293±0.523
E	0.616±0.600	0.433±0.570
F	0.650±0.667	0.553±0.678
Global	2.0789±1.743	1.733±1.765

Table 7: Perfomance of ViT single-stage model on Consensus and Brixia datasets in terms of MAE and STD

zone	MAE
	Senior and Junior
A	0.391±0.558
B	0.417±0.562
C	0.417±0.589
D	0.339±0.515
E	0.411±0.552
F	0.469±0.637
Global	1.943±1.777

Table 10: MAE and STD between senior and junior labels for Cohen dataset
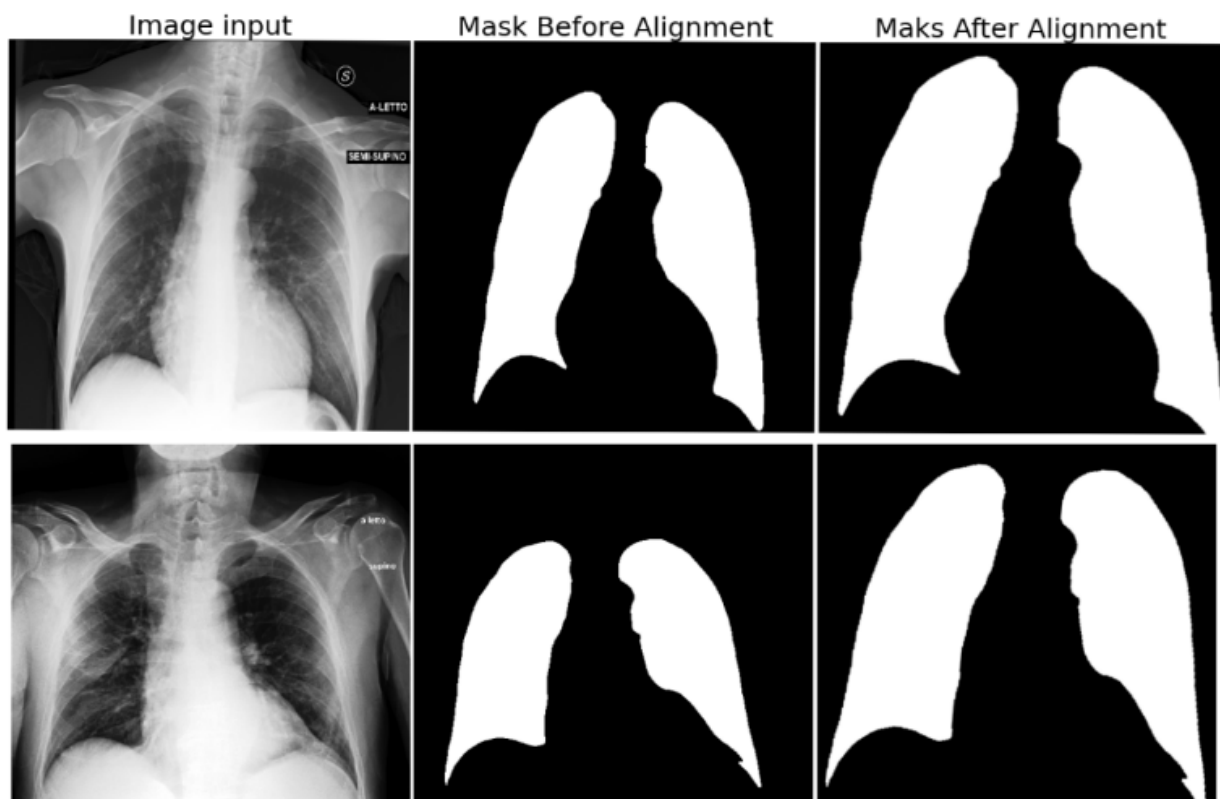
Figure 13: Segmentation mask before and after alignment using STN

Figure 14: confusion matrices of ViT_single-stage model(left), ViT multi-stage model(middle), and FPN-based model (right) for consensus dataset predictions on lung regions score values (top [0-3]) and on Global score values (bottom [0–18])

Figure 15: confusion matrices of ViT single-stage model(left), ViT multi-stage model(middle), and FPN-based model (right) for Cohen dataset predictions with respect to senior predictions

the classifier, and without STN on the FPN-based network. Finally, we compare the performance of single-stage ViT against multi-stage ViT and FPN-based networks on Brixia, Cohen, and consensus datasets to examine consistency and generalization of the different networks.

For the segmentation block, different architecture-encoder combinations were compared in terms of DSC and IoU on the segmentation test set and visually assessed on Brixia dataset. Attention U-Net and Attention ResU-Net had low DSC and IoU, so they were excluded from the rest of the experiments. As shown in table 4, all models have very similar performance with ResNet18+FPN, ResNet18+U-Net++, efficientnetb0+U-Net, and efficientnetb0+LinkNet producing slightly higher DSC and IoU values on the segmentation test set. So t-test was performed between the four models to test if the difference between them is significant. The t-test of the four models compared to each other was above 0.05. The performance of the four models was also visualized on the Brixia target dataset since it has more hard to classify examples compared to the segmentation dataset. Figure 11 illustrates the difference in performance among the four models in one of the hard to classify cases from Brixia dataset.

After the performance of segmentation block was analyzed, we started building the multi-stage network by combining segmentation, pretrained STN, and classification blocks as shown in table 5. Figure 13 is a visualization of the effect of applying transformation matrix from pretrained STN on the segmentation mask from Brixia dataset. For classification, We first started with a baseline ResNet18 classifier as all in one network. Then, segmentation, alignment, and FPN-based classifier were added one by one. Adding the segmentation block reduced MAE in some regions and increased it in other regions while adding alignment and FPN reduced the MAE in all regions except region F.

To test the effect of STN, Table 6 lists the MAE and STD of STN as a pretrained block and as a train-

able module with the classification block and without STN. Pretrained STN block outperformed STN trainable module and had lower MAE compared to without using any alignment in the network. The STN experiments were done using the FPN-based multi-stage network.

When the final FPN-based network was built, FPN was replaced with a ViT, but the MAE increased over all regions compared to the FPN-based model on Brixia dataset. So, ViT was added between the FPN and dense layer and MAE over all regions decreased compared to FPN-based model as shown in table 8 & 9.

To study the contribution of segmentation, alignment, and FPN to the ViT, a single-stage ViT model without alignment and segmentation modules was examined and the results are reported in table 7. The results were comparable with the multi-stage ViT on Brixia dataset and better on consensus subset.

To assess the inter-rater variability of the three developed pipelines, MAE and STD were evaluated on consensus dataset in which the labels are the majority voting(mode) of five radiologists. For better understanding of the inter-rater variability, MAE and STD between senior and junior labels for Cohen dataset were calculated and reported in table 10.

Figure 14 shows the confusion matrices of the three networks with regional scores predictions[0:3] at top and global scores predictions[0:18] at the bottom on the consensus dataset. FPN-based model showed the most correct and consistent predictions along the four scores. On the other hand, multi-stage ViT model had sparse predictions compared to single-stage ViT. As shown in tables 7, 8, and 9, although ViT models had better performance on Brixia dataset, FPN-based model surpassed ViT multi stage model and had comparable MAE with ViT single-stage model on the consensus dataset.

For testing generalization capabilities of the three models, models were evaluated on Cohen dataset. All models were more correlated with the senior predictions than the junior predictions, so only the confusion matri-

ces with respect to senior labels are displayed in figure 15. ViT single-stage had the most consistent predictions compared to multi-stage ViT and FPN-based networks; however, ViT multi-stage had more correct predictions for three scores compared to single-stage model. FPN –based model had more correct predictions on the extreme scores 0 and 3.

6. Discussion

In this study, clinical and deep learning knowledge were exploited for developing automated solution for brixia score predictions and testing robustness and generalization on different datasets. In this section, the clinical findings association with Brixia dataset distribution and deep learning models evaluation will be discussed first. Then, the performance of each block of the multi-stage model will be investigated. After that, performance of multi-stage and single stage models on cohen and consensus datasets will be analyzed.

The brixia distribution shown in figure 6 was consistent with the radiologists findings about the most diseases zones of COVID patient lungs. Different studies indicated that the lower parts of the lungs are more frequently involved and that the involvement is usually bilateral (Yasin and Gouda, 2020). In figure 6, zones C and F, which are the lower zones in Brixia system, has higher severity scores compared to the upper zones. Also, the bilateral pattern between bilateral zones scores distribution is clearly visible (e.g. zone A and D, upper zones have similar distribution). On the other hand, figure 7 and 8 are the distribution of Brixia scores for two board certified radiologists with different years of experience. The difference in score distributions, especially in regions D and F, affirm the inter-rater variability among radiologists in classifying lung severity. AI can play a crucial rule in having a more consistent scoring assessment; However, most of the algorithms are trained on datasets that lacks multireader assessment and that was the main purpose of evaluating our models with respect to the consensus and cohen datasets. During our discussion with Dr. Annemiek snoeckx, head of radiology department at Antwerp University Hospital, she said that radiologists are looking for a more objective and consistent quantification of lung opacity and that automated solutions will be very useful for pulmonary patients especially in the intensive care unit, ICU. Given that information, we believe that the best model is not only the one producing the lowest error, but also the one that shows a consistent performance among different datasets and along the six zones and four scores. To achieve the goal of automated lung opacity quantification, multi-stage and single stage models were developed and evaluated on Brixia, Cohen, and consensus datasets mentioned in section 3.

In multi-stage model, segmentation was the first step to extract the region of interest and the results of dif-

ferent backbone-encoder combinations of the state-of-the-art are shown in table 4. After analyzing segmentation results, it is found that the highest performing models in terms of DSC and IoU are ResNet18+FPN, ResNet18+U-Net++, efficientnetb0+U-Net, efficientnetb0+LinkNet; nevertheless , the statistical t-test for the four models with respect to each other was greater than 0.1 which means that they are falling in the same distribution and that the difference in DSC and IoU is not statistically significant. However, when the four models were visually evaluated on Brixia dataset which has more hard to segment examples, U-Net and LinkNet included regions outside the lung fields in their predictions as shown in figure 11, so the final model chosen was FPN since it has lower number of parameters compared to U-Net++ and has comparable visual masks.

For the alignment module, removing STN or using it as a differentiable layer increased the MAE. We assume that is because pertained STN is specifically trained for the task of aligning and zooming the lung masks as shown in figure 13 which is critical for pooling the correct six lung zones. In the classification module, utilizing ResNet18 backbone as the only feature extractor for the Brixia prediction task had the highest MAE on Brixia dtaset and hence it was excluded from rest of the experiments on consensus and cohen datasets. Adding pretrained STN lowered the MAE as it aligns and zooms into the ROI which is important for the pooling step. Adding FPN to ResNet18 backbone also lowered the MAE as FPN enhances the extracted feature maps exploiting the idea of multi-scale feature maps fusion. Using Backbone-ViT in multi-stage and single stage slightly lowered the MAE of the six regions on brixia dataset; however, the performance was very different with FPN model when tested on cohen and consensus datasets.

For having a more reliable reference for evaluation, models were tested on consensus dataset. ViT single-stage and FPN-based model outperformed the ViT multi-stage by a big difference in MAE which indicates that those two model are robust to different radiologists scoring on Brixia dataset.

When testing on Cohen dataset, the three models were agreeing more with the more experienced radiologist (senior: 20+ years of experience) in terms of a lower MAE and better confusion matrix. In addition, ViT-based models were more consistent along the four scores compared to the FPN-based model as shown in figure 15. FPN-based model had bias towards the extreme scores, 0 and 3.

Given that, we believe that ViT alone or combined with segmentation and alignment is more robust and consistent compared to CNN classifiers. Furthermore, ViT can eliminate the necessity for segmentation and alignment which is useful in having one stage training and optimization. It also converges faster compared to the multi-stage model which is adding more value in re-

ducing the training time.

An interesting future work to investigate is to pretrain transformers with large CXR dataset (e.g. cheXNet), fine-tune it with the Brixia dataset, and evaluate it on the three datasets used in this research. Another interesting approach is to evaluate the models designed for Brixia score on other scoring systems dataset (e.g. RALE dataset) and investigate the correlation between the two scoring systems.

7. Conclusions

The motivation of this research was the strong clinical need of a consistent, automated solution for lung severity assessment. The availability of large dataset of 5,000 annotated CXRs and other small datasets annotated by several radiologists helped in analyzing the inter-rater variability and generalization problem. In addition, Brixia dataset allowed to apply data hungry architectures like ViT. Three different models were developed and compared against each other in terms of consistency and robustness on different datasets. FPN-based multi-stage model consisted of segmentation using FPN, alignment using STN, and classification using another FPN. IN the second model, ViT multi-stage, ViT was inserted between FPN and dense layer and everything else remained fixed. The third model was a single-stage ViT model in which ResNet backbone was combined with ViT and trained at once as a single unit. The results from the ViT models on the three different datasets were promising and consistent which is encouraging for more investigations about how ViT can aid radiologists with more generalized and consistent automated scoring systems.

8. Acknowledgments

References

Borghesi, A., Maroldi, R., 2020. Covid-19 outbreak in italy: experimental chest x-ray scoring system for quantifying and monitoring disease progression. La radiologia medica 125, 509–513.

Bradley, S.H., Abraham, S., Callister, M.E., Grice, A., Hamilton, W.T., Lopez, R.R., Shinkins, B., Neal, R.D., 2019. Sensitivity of chest x-ray for detecting lung cancer in people presenting with symptoms: a systematic review. British Journal of General Practice 69, e827–e835.

Çallı, E., Sogancioglu, E., van Ginneken, B., van Leeuwen, K.G., Murphy, K., 2021. Deep learning for chest x-ray analysis: A survey. Medical Image Analysis 72, 102125.

Chamorro, E.M., Tascón, A.D., Sanz, L.I., Vélez, S.O., Nacenta, S.B., 2021. Radiologic diagnosis of patients with covid-19. RadiologíA (English Edition) 63, 56–73.

Chaurasia, A., Culurciello, E., 2017. Linknet: Exploiting encoder representations for efficient semantic segmentation, in: 2017 IEEE Visual Communications and Image Processing (VCIP), IEEE. pp. 1–4.

Chen, X., Diaz-Pinto, A., Ravikumar, N., Frangi, A.F., 2021. Deep learning in medical image registration. Progress in Biomedical Engineering 3, 012003.

Cohen, J.P., Dao, L., Roth, K., Morrison, P., Bengio, Y., Abbasi, A.F., Shen, B., Mahsa, H.K., Ghassemi, M., Li, H., et al., 2020. Predicting covid-19 pneumonia severity on chest x-ray with deep learning. Cureus 12.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

Gaál, G., Maga, B., Lukács, A., 2020. Attention u-net based adversarial architectures for chest x-ray lung segmentation. arXiv preprint arXiv:2003.10304 .

Gordienko, Y., Gang, P., Hui, J., Zeng, W., Kochura, Y., Alienin, O., Rokovyi, O., Stirenko, S., 2018. Deep learning with lung segmentation and bone shadow exclusion techniques for chest x-ray analysis of lung cancer, in: International conference on computer science, engineering and education applications, Springer. pp. 638–647.

Gour, M., Jain, S., 2022. Automated covid-19 detection from x-ray and ct images with stacked ensemble convolutional neural network. Biocybernetics and Biomedical Engineering 42, 27–41.

Harahwa, T.A., Yau, T.H.L., Lim-Cooke, M.S., Al-Haddi, S., Zeinah, M., Harky, A., 2020. The optimal diagnostic methods for covid-19. Diagnosis 7, 349–356.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hesamian, M.H., Jia, W., He, X., Kennedy, P., 2019. Deep learning techniques for medical image segmentation: achievements and challenges. Journal of digital imaging 32, 582–596.

Homayounieh, F., Zhang, E.W., Babaei, R., Karimi Mobin, H., Sharifian, M., Mohseni, I., Kuo, A., Arru, C., Kalra, M.K., Digumarthy, S.R., 2020. Clinical and imaging features predict mortality in covid-19 infection in iran. Plos one 15, e0239519.

Jaderberg, M., Simonyan, K., Zisserman, A., et al., 2015. Spatial transformer networks. Advances in neural information processing systems 28.

Jaeger, S., Candemir, S., Antani, S., Wáng, Y.X.J., Lu, P.X., Thoma, G., 2014. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. Quantitative imaging in medicine and surgery 4, 475.

Jiang, Z.z., He, C., Wang, D.q., Shen, H.l., Sun, J.l., Gan, W.n., Lu, J.y., Liu, X.t., 2020. The role of imaging techniques in management of covid-19 in china: from diagnosis to monitoring and follow-up. Medical science monitor: international medical journal of experimental and clinical research 26, e924582–1.

Kumar, V., Zarrad, A., Gupta, R., Cheikhrouhou, O., 2022. Cov-dls: Prediction of covid-19 from x-rays using enhanced deep transfer learning techniques. Journal of Healthcare Engineering 2022.

Kundu, S., Elhalawani, H., Gichoya, J.W., Kahn Jr, C.E., 2020. How might ai and chest imaging help unravel covid-19's mysteries? Radiology: Artificial Intelligence 2.

Lewis, P., Czum, J.M., 2013. Chest imaging. Oxford American Handbook of Radiology , 41.

Li, H., Liu, B., Zhang, Y., Fu, C., Han, X., Du, L., Gao, W., Chen, Y., Liu, X., Wang, Y., et al., 2021. 3d ifpn: Improved feature pyramid network for automatic segmentation of gastric tumor. Frontiers in Oncology 11, 1654.

Li, M.D., Arun, N.T., Gidwani, M., Chang, K., Deng, F., Little, B.P., Mendoza, D.P., Lang, M., Lee, S.I., O'Shea, A., et al., 2020. Automated assessment and tracking of covid-19 pulmonary disease severity on chest radiographs using convolutional siamese neural networks. Radiology: Artificial Intelligence 2.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017a. Feature pyramid networks for object detection, in: Pro-

ceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017b. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Litmanovich, D.E., Chung, M., Kirkbride, R.R., Kicska, G., Kanne, J.P., 2020. Review of chest radiograph findings of covid-19 pneumonia and suggested reporting language. Journal of thoracic imaging 35, 354–360.

Lomoro, P., Verde, F., Zerboni, F., Simonetti, I., Borghi, C., Fachinetti, C., Natalizi, A., Martegani, A., 2020. Covid-19 pneumonia manifestations at the admission on chest ultrasound, radiographs, and ct: single-center study and comprehensive radiologic literature review. European journal of radiology open 7, 100231.

Maroldi, R., Rondi, P., Agazzi, G.M., Ravanelli, M., Borghesi, A., Farina, D., 2021. Which role for chest x-ray score in predicting the outcome in covid-19 pneumonia? European Radiology 31, 4016–4022.

Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., et al., 2018. Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999 .

Omar, S., Motawea, A.M., Yasin, R., 2020. High-resolution ct features of covid-19 pneumonia in confirmed cases. Egyptian Journal of Radiology and Nuclear Medicine 51, 1–9.

Pahiju, M., Thapa, N., 2017. Findings of chest radiographs of opd patients in tu teaching hospital. Journal of Institute of Medicine 40, 75–9.

Park, S., Kim, G., Oh, Y., Seo, J.B., Lee, S.M., Kim, J.H., Moon, S., Lim, J.K., Ye, J.C., 2021. Vision transformer for covid-19 cxr diagnosis using chest x-ray feature corpus. arXiv preprint arXiv:2103.07055 .

Pham, T.D., 2020. A comprehensive study on classification of covid-19 on computed tomography with pretrained convolutional neural networks. Scientific reports 10, 1–8.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Setiawati, R., Widyoningroem, A., Handarini, T., Hayati, F., Basja, A.T., Putri, A.R.D.S., Jaya, M.G., Andriani, J., Tanadi, M.R., Kamal, I.H., 2021. Modified chest x-ray scoring system in evaluating severity of covid-19 patient in dr. soetomo general hospital surabaya, indonesia. International Journal of General Medicine 14, 2407.

Signoroni, A., Savardi, M., Benini, S., Adami, N., Leonardi, R., Gibellini, P., Vaccher, F., Ravanelli, M., Borghesi, A., Maroldi, R., et al., 2021. Bs-net: Learning covid-19 pneumonia severity on a large chest x-ray dataset. Medical Image Analysis 71, 102046.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

Souza, J.C., Diniz, J.O.B., Ferreira, J.L., da Silva, G.L.F., Silva, A.C., de Paiva, A.C., 2019. An automatic method for lung segmentation and reconstruction in chest x-ray using deep neural networks. Computer methods and programs in biomedicine 177, 285–296.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826.

Tahir, A.M., Qiblawey, Y., Khandakar, A., Rahman, T., Khurshid, U., Musharavati, F., Islam, M., Kiranyaz, S., Al-Maadeed, S., Chowdhury, M.E., 2022. Deep learning for reliable classification of covid-19, mers, and sars from chest x-ray images. Cognitive Computation , 1–21.

Tan, M., Le, Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR. pp. 6105–6114.

Wang, L., Lin, Z.Q., Wong, A., 2020a. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. Scientific Reports 10, 1–12.

Wang, Y., Wang, Y., Chen, Y., Qin, Q., 2020b. Unique epidemiological and clinical features of the emerging 2019 novel coronavirus pneumonia (covid-19) implicate special control measures. Journal of medical virology 92, 568–576.

Warren, M.A., Zhao, Z., Koyama, T., Bastarache, J.A., Shaver, C.M., Semler, M.W., Rice, T.W., Matthay, M.A., Calfee, C.S., Ware, L.B., 2018. Severity scoring of lung oedema on the chest radiograph is associated with clinical outcomes in ards. Thorax 73, 840–846.

Wong, A., Lin, Z., Wang, L., Chung, A., Shen, B., Abbasi, A., Hoshmand-Kochi, M., Duong, T., 2021. Towards computer-aided severity assessment via deep neural networks for geographic and opacity extent scoring of sars-cov-2 chest x-rays. Scientific reports 11, 1–8.

Yasin, R., Gouda, W., 2020. Chest x-ray findings monitoring covid-19 disease course and severity. Egyptian Journal of Radiology and Nuclear Medicine 51, 1–18.

Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J., 2018. Unet++: A nested u-net architecture for medical image segmentation, in: Deep learning in medical image analysis and multimodal learning for clinical decision support. Springer, pp. 3–11.

Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J., Zhao, X., Huang, B., Shi, W., Lu, R., et al., 2020. A novel coronavirus from patients with pneumonia in china, 2019. New England journal of medicine .

Zimatore, C., Pisani, L., Lippolis, V., Warren, M.A., Calfee, C.S., Ware, L.B., Algera, A.G., Smit, M.R., Grasso, S., Schultz, M.J., 2021. Accuracy of the radiographic assessment of lung edema score for the diagnosis of ards. Frontiers in physiology 12.

Towards understanding of facial nerve stimulation in cochlear implant patients with automatic transformer pipeline

Muhammad Roshan Mughees[a], Raabid Hussain[a], Paula Lopez Diez[c], Octavio Martinez Manzanera[a], François Patou[b], Jan Margeta[d]

[a]*Oticon Medical, 14 Chemin de Saint-Bernard Porte, 06220 Vallauris, FRANCE*
[b]*Oticon Medical, Research & Technology group, Smørum, DENMARK*
[c]*DTU Compute, Technical University of Denmark, Kongens Lyngby, DENMARK*
[d]*KardioMe, Research and Development, Nová Dubnica, SLOVAKIA*

Abstract

The facial nerve (FN) is not only important for cochlear implant (CI) surgery, but it is also one of the most well-known and common concerns of the treatment, with Facial Nerve Stimulation (FNS) possibly occurring as a result of its closeness to some of the implant's electrodes. FN is responsible for controlling expressions and facial movements so when this nerve is stimulated because of the proximity to the electrodes then it can cause temporary or permanent damage to expressions. Knowing the FN location, as it passes through the cochlea structure might help prevent this stimulation, which can cause severe involuntary motion. The detection and segmentation of the FN is a complex and time-consuming procedure, also, measuring its closeness to the electrodes based on preoperative computed tomography (CT) scans automatically has not been done before. The absence of contrast in CT scans makes the neural structures look extremely similar to other types of tissue. We propose an automatic pipeline of segmenting the cochlea and facial nerve from the preoperative CT scans. We then find the distances of electrodes to the nearest point of the facial nerve in order to identify and prevent facial nerve stimulation (FNS).

Keywords: Facial Nerve Stimulation, 3D Image Segmentation, Cochlear Implant

1. Introduction

Hearing is an important ability for humans. As social beings, we communicate with one another, which is important for our proper growth. The universe of experience can be significantly hampered by partial injury or an absence of this sense. The most common sensory deficiency in humans is hearing loss. Sensorineural hearing loss (SHNL) is the most prevalent kind of hearing loss in adults, accounting for over 90% of all occurrences of hearing loss (Smith et al. (2005)). Cochlear implants (CI) were developed to help people with SNHL who have lost their hearing abilities due to congenital or acquired causes.

A cochlear implant is a medical device (also developed by Oticon Medical) that helps restore the hearing capacity of people who suffer sever to profound. It helps to send signals to the brain by stimulating the auditory nerve in the cochlea. An external sound processor and an inside cochlear implant make up a cochlear implant. The antenna is magnetically connected to the skin immediately above the internal section. The exterior part includes a behind-the-ear sound processor and a lead that links the processor to the antenna. Internally, a receiver is surgically slipped beneath the skin on the temporal bone. The electrode array in the cochlea is part of the receiver.

A conventional cochlear implant procedure starts by making a small incision behind the ear, drilling out a portion of the mastoid bone (mastoidectomy), carefully avoiding critical structures such as facial nerve, chorda tympani and vessels until the round window is revealed and inserting the electrode array into the scala tympani via either round window or cochleostomy. Oticon Medical has developed an image analysis tool that extracts clinically relevant information about the cochlea that is useful to determine the optimal surgical approach

and electrode array, and to reduce trauma during insertion. The route must pass via a 1-3.5 mm area, near the branching of the FN and chorda tympani. An illustration of this procedure is shown in Figure1. Damage to this nerve might result in facial paralysis (Noble et al. (2008)).
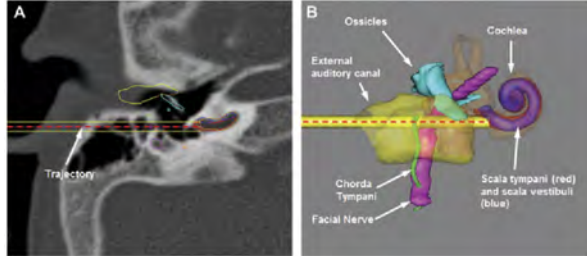
Figure 1: Representation of cochlear implant in CT on left and structural representation on right

Not only is the placement of this nerve critical for implant surgery, but it is also one of the most well-known and common concerns of the CI procedure: FN activation may be a result of its closeness to part of the implant's electrodes (see Figure 2) as literature has suggested.
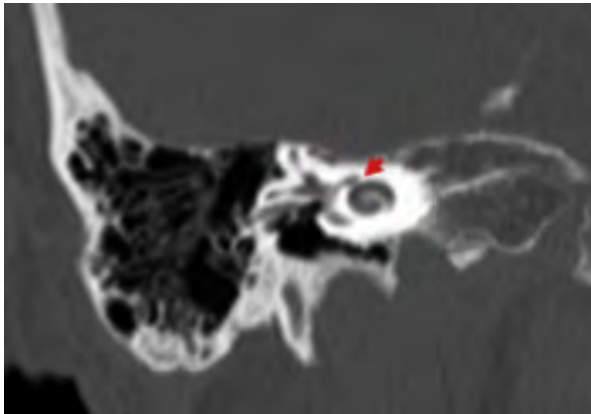
Figure 2: In high-resolution temporal bone CT scan of a patient, where he reported reported FNS (eye twitching) caused by electrodes 9-13 (Fang et al. (2017))

Knowing the exact location of the FN and calculating the distance from cochlea before surgery might help us prevent it from happening. Even for specialists, locating and segmenting the nerve is challenging due to its curved form and lack of contrast, which makes the nerve tissues look quite similar to soft tissue structures. We aim to develop and implement methods capable of automatically segmenting the structures of interest and finding their distance from electrodes. The electrodes are placed during the surgery. The distance between FN and electrodes can be used to prevent FNS (Polak et al. (2006)) by proper surgical planning i.e., adjusting the intensity of electrodes during the fitting procedure. Facial nerve stimulation can frequently be re-

solved with minimal changes in speech processor fitting but, in some cases, this can lead to a reduction in the outcome.

1.1. Clinical background

Hearing involves sound waves traveling through our ears. Our ears are complex structures having three parts. The inner ear is the focus of our investigation. Both of the structures of interest (FN and Cochlea) are found in this area to some extent and some part in middle ear. These structures, their architecture, and their significance will be discussed in the following sections.

1.1.1. Facial Nerve

The seventh Cranial Nerve is the facial nerve according to its location, from the front to the back of the brain. The FN's route is complicated; there are several branches that provide sensory, motor, and parasympathetic information. It is responsible for:

- Facial expressions
- Chorda tympani which originates from it controls taste
- Salivation
- Eyelid closure
- Auditory reflex

The FN's path (Figure 3) may be divided into two parts: intracranial and extracranial. After entering the

Figure 3: Facial Nerve structure and shape (Campbell (2020))

inner ear the proximity between the FN and certain structures of the ear as the cochlea, vestibule or staples can be observed in the different views (Figure 4).
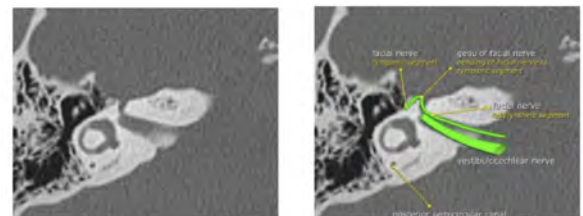
Figure 4: On right we can see the location of the facial nerve and on the left other structures in CT scan (Beek and Pameijer (2020))

The roots escape the internal auditory meatus within the temporal bone and enter the facial canal. The two

roots converge in this area to produce the FN, which gives rise to the geniculate ganglion (genu) (see Figure 5). The larger petrosal nerve, which controls the lacrimal gland and the mucous glands, emerges from this ganglion. The chorda tympani is formed by the FN and contains sensory fibers for the tongue's anterior two-thirds as well as parasympathetic fibers for the submandibular and sublingual glands.
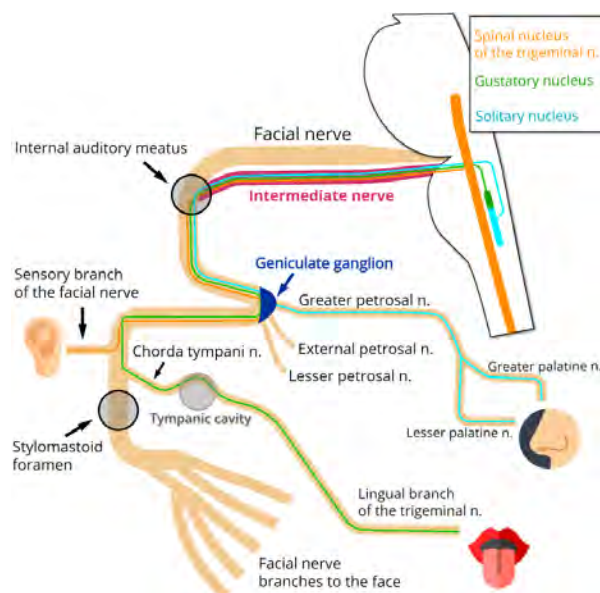
Figure 5: Internal anatomy of ear (de Castro and Marrone (2021))

1.1.2. Cochlea

Cochlea is also another important structure in hearing (Elliott and Shera (2012), Baker (2008)). The cochlea is snail-shaped organ present in the temporal bone that is about 8-10 mm wide and has 2.5 turns in humans typically. Because of its contrast and snail-shaped structure (Figure 2), it is relatively easy to segment and spot even while manually annotating the structures. In fact while manually annotating other structures, it can be a guiding tool for finding those other structures (see Figure 6).

Figure 6: Inner ear structures in clinical CT. In this blue arrow, we see the cochlea, the red arrow shows semicircular canals and green shows ossicles

2. State of the art

The literature on the FN, Cochlear segmentation, and measurement in clinical CT scans was reviewed and summarized in this section.

2.1. Segmentation

Previous research on facial nerve was mostly focused on atlas-based segmentation, deformable models, a mix of these two approaches, and segmentation based on landmark predictions. A summary of these methods can be seen on Table 1.

In early study, Noble et al. (2008) offered the first automated atlas-based strategy, which merged atlas methodology and a minimal cost pathfinding algorithm. The atlas was utilized to generate a spatially varying cost function that included geometric information, which was then used to determine the nerve centerline's minimal cost route. The entire structure was extracted with this route as seed. Result shows that atlas-based methods alone are ineffective. Mainly because the registration transformations deform the structures more than what is physically possible because they are very elastic. Due to fluctuations in pixel values in the neuronal structure and a lack of contrast, pathfinding algorithms relying only on intensity didn't work as well in this paper.

Then Voormolen et al. (2012) created a semiautomatic segmentation technique for a larger part of the FN in CT called NerveClick. Here an expert surgeon is needed to manually set two markers in specific positions of the FN anatomy. They also created a statistical and texture model based on the centerline of 40 manually segmented examples' facial nerves. They iteratively deform the model using this model and the manual landmarks as seeds until the convergence conditions are met. Firstly, this approach still required external input for initialization and still when applied in patients with cranial base tumors and/or severely disturbed temporal bone structure, the statistical model was not strong enough, resulting in worse performance. 26 percent of the centerlines in their testing set were deemed to be off-center.

Powell et al. (2017) created an atlas-based automatic segmentation system that included the cochlea, ossicles, and semicircular canals, and other structures from the temporal bone CT scans. The atlas and the several ROIs that surround each anatomical feature were developed using six bones. The FN was segmented using three distinct masks, one in the tympanic area and the other two in the mastoid region, following registration. Then they eroded the segmentation by one voxel in each slice and followed the segmented object with the closest centroid along the length of the FN. The performance was still limited because of the FN's shape and poor contrast in CT scans, especially in the tympanic area.

The first deep learning approach was used by Fauser et al. (2019) where they used a shape-regularized deep learning approach for segmenting small structures based on the anatomy. Like us, they were also segmenting these structures for trajectory planning in CI operations. Instead of utilizing the 3D nature of CT scans, they used 2D slices in their method. So the predictions done were for each slice of a specific view. Multiple U-Nets predictions were merged to create the initial segmentation. The results showed a lot of artifacts and missing segmentations mainly because of the class imbalance (as these are small structures). Probabilistic active shape models (PASM) were created to address these challenges and give some 3D form regularization. This made it more flexible but they only focussed on some part of FN which is important in the surgery of Cochlear Implant.

Then Gare et al. (2020) created a multi-atlas-based FN segmentation using nine samples from different segmentations. These samples were chosen in a way to maximize the variance between individuals. As a result, this method segmented additional portions of the FN. However, still here the expert has to manually set four landmarks in exact places making it difficult to segment automatically. Based on these landmarks, a centerline is created, which is then used to select the optimal registration from the atlas. Then the results may be fine-tuned later using B-splines registration and deformation limits.

López Diez et al. (2021) worked on segmentation of FN and cochlear nerve in pre-op CT Scans of patients. Their pipeline had two main stages: the prediction of seven landmarks by reinforcement learning from CT scan and then using shortest path algorithm Dijkstra to join those landmarks for segmenting the structures. They did this by first annotating the dataset with seven critical landmarks and succeeded in locating the landmarks in the CT scans yielding to 96.10% of correctly located landmarks in the test set. However, when there are changes in anatomy, such as between children and adults, the approach may not perform well as shown in Reda et al. (2011). Also, there was inconsistency in placement of the landmarks because it is just a point on a straight-ish curve and suffers from the aperture problem as shown in Figure 7

Figure 7: Landmarks' location within the FN and CN. A) An overview of the 7 landmarks B) Close-up of the 4 initial landmarks and the labyrinthine segment C) Close-up of the 5-7 landmarks and the tympanic segment (López Diez et al. (2021))

piction since they have a consistent form. This is true of brain regions like the liver and kidneys.

Its segmentation from CT images of the temporal bone is difficult due to the low resolution of those images in comparison to the anatomy of the cochlea: the cochlea dimensions are about 8.5x7x4.5 mm³, whereas the typical CT voxel size is larger than 0.2 mm, making the fine structures of the chambers barely visible. In addition, the cochlea is filled with fluids that are similar in appearance to those nearby structures on CT scans. In many circumstances, Deep Learning is an effective method of picture segmentation or processing. Many studies in the field of inner ear CT imaging analysis have shown outstanding findings (Wang et al. (2021), Banalagay et al. (2021), Lv et al. (2021), Hussain et al. (2021), Nikan et al. (2020), Neves et al. (2021), Heutink et al. (2020), Zhang et al. (2019), Ruiz Pujadas et al. (2018), Demarcy (2017), Gerber et al. (2017), Kjer et al. (2015), Noble et al. (2011), Abeysinghe et al. (2008)), but each had a number of drawbacks. To begin with, developing dataset annotations takes time, which may limit the production of large training datasets. A well-trained ENT surgeon would take at least 10 minutes to segment each 3D cochlea volume in the instance of the cochlea. These methods perform well and some use shape models for further regularization. Since they are not using a common dataset, their direct comparison is challenging. We summarize their performance in Table 2.1. In this thesis, we show that significant improvements can be obtained using more recent network architectures.

2.2. Distance calculation

Currently, there is no automatic pipeline of measuring the distance from the facial nerve to the electrodes or the cochlea before or after insertion, measurement calculation regarding the CT images to the best of our knowledge. Only method used by Jonathan (Hatch et al. (2017)) was done manually. To the best of out knowledge, this is the first work that automatically computes the FN nerve distance to the CI and is the primary focus

Year	Authors	Method	Inputs	Training size	Testing size	Region Segmented
2008	Nobel	Atlas + Deform. models	10 param	12CT (15 ears)	7CT (10 ears)	Facial Recess
2012	Voormolen	Deform. models	2 Landm.	40 CT	120 CT	IAC to SF
2017	Powell	Atlas-based	None	6	20	IAC to SF
2019	Fauser	DL+PASM	None	Not specified	24 CT	IAC to 2nd Genu
2020	Gare	Multi-Atlas	4 Landm.	37 micro CT(dead)	28 CT (alive)	IAC to SF
2021	Paula	Reinforcement learning + Dijkstra	None	96 CT	23 CT	Cochlear Nerve + Facial Nerve

Table 1: Comparison of FN segmentation techniques with their respective datasets

A medical specialist may quickly distinguish certain anatomical structures from their three-dimensional de-

of this thesis. These measurements can help us reduce the chance of facial nerve stimulation during or after the CI surgery.

3. Material and methods

For making this work, we use a substantial amount of material and we evaluate a large number of methods. Following are the details.

The overall processes taken throughout this thesis are summarized below with a flowchart and the details of each step are afterwards mentioned.
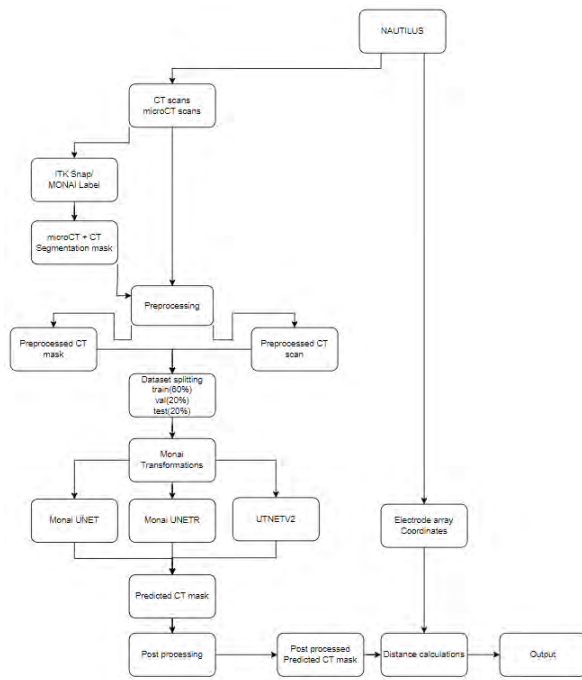
Figure 8: Pipeline of our evaluation methodology

3.1. Dataset

A dataset's size, diversity and quality are critical for Deep Learning DL techniques, since they represent the backbone of the training process and they are hence critical to the model's success. A dataset with temporal bone structures labeled in good resolution is hard to find. Two datasets were mainly used. One dataset which was openly available. OpenEar (Sieber et al. (2019)) contains 3D models of temporal bone structures based on CT and micro-slicing. And the other was the data provided by Oticon Medical. The OpenEar dataset is utilized for familiarization and anatomical detail analysis, whereas the Oticon Medical data set has been labeled and divided for training, validation, and testing. This Oticon Medical dataset is created by merging datasets from different hospitals and from different countries. There was no labeled data with important information on the FN or the other structures accessible

for the development of this project. As a result, the first step of this project was annotating a significant number of examples from the data. During the data analysis, this was taken into consideration.

3.1.1. OpenEar Dataset

The dataset (Sieber et al. (2019)) contains high quality coloured models of the human temporal bone (see Figure 9). This dataset is publicly available with 3D models of structures for robotic surgery, development of segmentation algorithms etc.

Figure 9: On the left there is the CBCT scan, the segmentation of it in the middle and on right there is the 3D model of the structures (Sieber et al. (2019))

This dataset was created using 8 human temporal bone specimens from four adult participants. The Hannover Medical School's Institute of Pathology generously supplied temporal bones. Each of the scans had good resolution with 0.125 mm voxel spacing. The dimensions varied a lot in each of the scans from 150-850 also across different axis (see Figure 10 for an example). Also the contrast ranges were from -2500 to 4000 Hounsfield units which also needed adjusting. Although, it is opensource but even this dataset had problems in some scans. It shows that we have to do some preprocessing to remove these artifacts. The names of
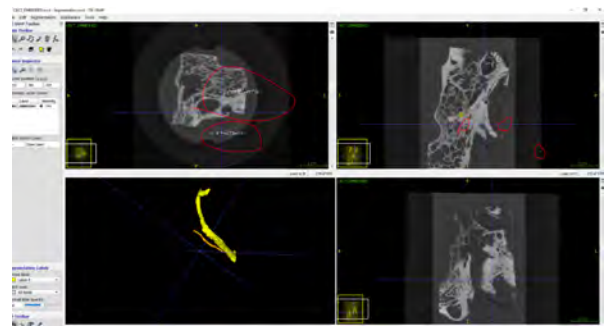
Figure 10: Example from the OpenEar dataset as being annotated in ITK-SNAP for facial nerve and chorda tympani

these are given on mathematical symbols like alpha, beta, theta as all the metadata related to the patient was stripped from this dataset.

3.1.2. OticonMedical dataset

CT images of patients before surgery from the Oticon Medical database are used. It contained 80 CT scans of different patients' inner ear (including right or left labeled). The CTs have been cropped in the area of interest and come from a variety of imaging systems from

different hospitals from different countries and well represent the real-world clinical image appearance diversity. As a result, the dataset is typical of patients with normal anatomy and real-world imaging to which clinicians have access for diagnosis and intervention planning.

3.1.3. Data challenges of the OticonMedical dataset

Here the range of scans are similar to what other relevant studies but there are several challenges that we face.

- The resolution of these CT scans is not the same in all the images (see Fig. 11).
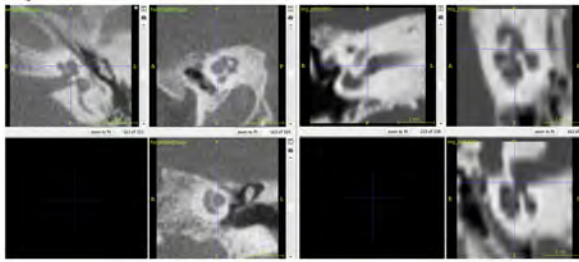
Figure 11: Left and Right CT scans showing different resolutions

- Example from the OpenEar dataset as being annotated in ITK-SNAP for facial nerve and chorda tympani.

- Some images have a considerable large amount of noise and some images have large invalid regions.

- There is also a considerable difference in tissue contrast (Figure 12), owing to the images' varied origins (imaging equipment).
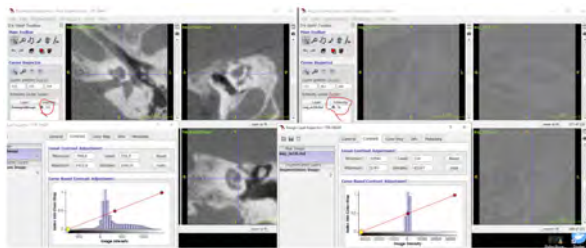
Figure 12: Differences in same tissue contrast values and their histograms. Left part of image shows is one 3D volume from our dataset and on right is another. Both have different intensity value for cochlea area.

- There are also scans with various forms of distortions (see Figure 13).

3.1.4. Annotating the dataset - OticonMedical

So even with all these challenges we annotated the dataset to generate the images and the segmentation
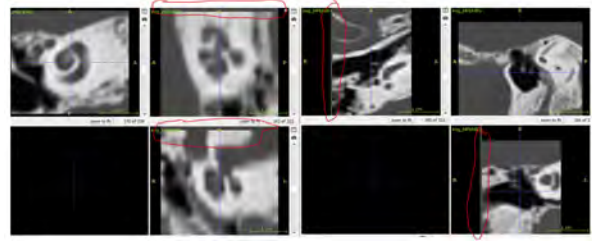
Figure 13: Artifacts and distortions present in original scans due to low resolution thus, preventing accurate delineation of small structures

masks together. For annotating the dataset, we used ITK-SNAP (Yushkevich et al. (2006)). It is a software for 3D medical image visualization and segmentation as it has some functions which are very useful for it. The paintbrush tool was used to place "seeds". These seeds are the points which belonged to a respective class and that will guide the segmentation in its favor. The labels were chosen using Quick Label Picker and then the marks were placed. (see Figure 14) After placement, this tool provides a 3D view that continuously updates the segmentation and provides a visualization of the nerves structures.
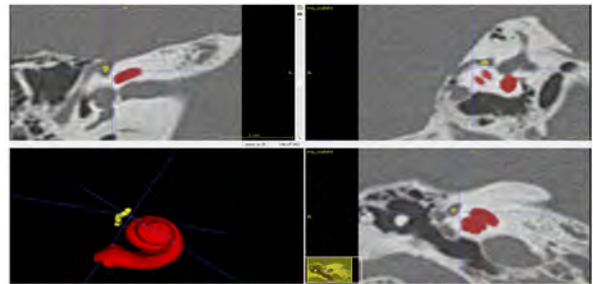
Figure 14: After placement of seed points for facial nerve

We then used Active Contour tool (Yushkevich et al. (2006)) for filling in the empty space from seeds (see Figure 15). First, we set a lower and upper threshold depending on the image intensity values to be more exact. The seeds that we initialized can be used as seeds for ac-
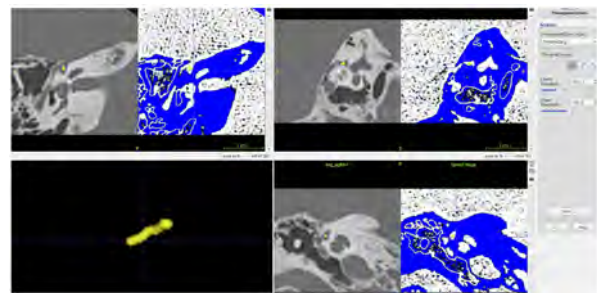
Figure 15: Contrast threshold to use specific region. Blue color represents the region to exclude and white represents the region to include

tive contour and we run it for some iterations (typically

6-10). This way we could get a rough segmentation of the facial nerve and then we modify it by adjusting the boundaries according to our knowledge. We used this process to annotate 70 images by first checking if they were in good quality and their FN is clear enough for us to segment.

3.2. Preprocessing of datasets

To address the above mentioned challenges, we used several strategies which are illustrated on Figure 16.

3.2.1. Preprocessing Pipeline
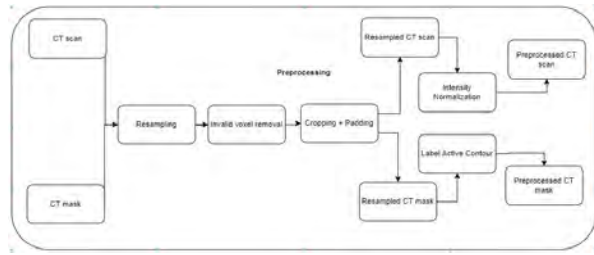
Figure 16: Preprocessing pipeline of CT scans

In summary, the obtained CT scans and annotated masks are resampled to make their spacing equal to each other. Then removed the artifacts to some extent from the images and similarly the same portion of image from the masks (even though it can be all background) removing invalid pixels. To make their sizes similar, applied padding and/or cropping to the images and masks alike. Then images are normalized to make their intensity range to 0-1 range and active contour is applied to labels to make them smooth.

3.2.2. Resampling

Some scans had significant anisotropic spacing across the three coordinates. Also, it was observed that very few samples also had really high spacing at .3 mm for all three coordinates. Apart from new spacing, the images were geometrically identical. To make it consistent, the scans were resampled with linear interpolation using SimpleITK (Johnson et al. (2013)) while the masks were interpolated with nearest neighbour technique. All the rest of the information like direction, origin were kept in accordance with the input images.

3.2.3. Label Active Contour

Due to interpolation, the labels now had block-like structure which is not good for training segmentation models. So for that in ITK-SNAP, Yushkevich et al. (2016), use active contour to make it more smooth around the corners and more similar to the real structure.

3.2.4. Invalid Voxel Removal

Clinical CTs contain often invalid voxels marked with large negative intensity regions at the outer space around the true image content. We pick values at the image corners and estimate the invalid voxel mask. This is done by first estimating the invalid voxel value (the minimum intensity of all 8 corners of the image) and region growing from all of them at the same time. This way, the artifacts with invalid values in capturing the CT scans can be removed. The function here returns a mask and we can take out the good part of the image from it.

3.2.5. Cropping (from segmentation center point)

Scans had different sizes (and simplify batching for the DL algorithms) we chose a fixed size of 256 voxels across all three axes. The correct part of the scan still had inconsistencies depending on the scan. So using the segmentation, we found out the centers of these images and cropped them accordingly to the size.

3.2.6. Intensity rescaling

The intensities of the CT scans can range from -3000 to +4000 Hounsfield Units (HU) or even more. Not all values in these range are useful for us. In fact, the intensities on the upper and lower bound are mostly irrelevant for our application. Implants made from metallic components typically have very high Hounsfield units (typically significantly greater than 4000) which we will see later in post-op images to be extremely bright. HU are obtained with reference to attenuation coefficient of water and divide by same value of water for normalization. Later this value is multiplied by a thousand resulting in huge values of intensity as observed by us. So first we clipped the scans from lower 2% to upper 98%. We used percentages because every scan had different range so it should be dynamic. And then we normalized these intensities so the values range from 0 to 1.

3.2.7. Padding

After intensity rescaling, there were some images where the size was less than our given size so we applied zero padding to them to make their size equal to the others and usable.

After all this, we had 70 scans out of which 40 are for training and 15 each for validation and testing.

3.2.8. Transformations

For data augmentation, we applied some transformations to our dataset. We used the MONAI framework (Diaz-Pinto et al. (2022)) for our pipeline of data augmentation. As we are well aware that contrast is very important in CT scans so we used transformations that made changes while keeping contrast almost constant. All the transformations we used were the dictionary transformations provided by MONAI. They are much easier to implement than others but require a specific

format to be followed for them to work. They are good as we can apply some transformations to both image and its mask to keep consistency.

After experimentation, we end up with using transformations such as:

- Random affine transform with 20 degree rotations, scaling 0.1 and translations with probability of 0.2

- Random flip along axial, sagittal and coronal axis separately with 0.2 probability

- Resized with interpolation nearest area for CT scan

- Resized with interpolation nearest neighbour for CT segmentation mask

3.3. Technical Background

In our approach we used deep learning for segmentation of the structures because other relevant research shows state of the art to be achieved through deep learning given enough data. For deep learning we used some models with pure CNN and some which were a combination of CNN and transformer architectures and will compare them. Each of them have a lot of differences in all their architectures.

3.3.1. Network Architecture

When it comes to medical image segmentation, U-Net (Ronneberger et al. (2015)) is one of the most commonly used and best-performing architectures, as it performs state of the art, or near, in most applications. It was developed with the goal of performing biomedical (microscopy) image segmentation. U-Net is a convolutional neural network that has two parts which are an encoder and decoder. The encoder compresses and extracts features from the image, arriving at the last layer having vector representation. The decoder reconstructs the segmented image through a series of up-scaling layers. It also has concatenation layers which help in this reconstruction process. The main feature of this architecture is that the concatenations that happen are on the same level, i.e., between the encoder and the decoder, as it can be seen on figure 17

3.3.2. UNEt TRansformers (UNETR)

Transformers achieved great success in NLP because of long-range sequence learning. Despite its effectiveness, purely convolutional neural networks' capacity to learn long-range spatial relationships is limited by their number of layers and their receptive field.

Like the Vision Transformers, UNETR (Hatamizadeh et al. (2022)) employs a transformer-based encoder but the decoder is CNN based. UNETR uses a transformer as the encoder. This encoder then learns sequence representations given the 3D medical scan and successfully captures global multi-scale information. Then skip connections are used to connect the encoder to the decoder
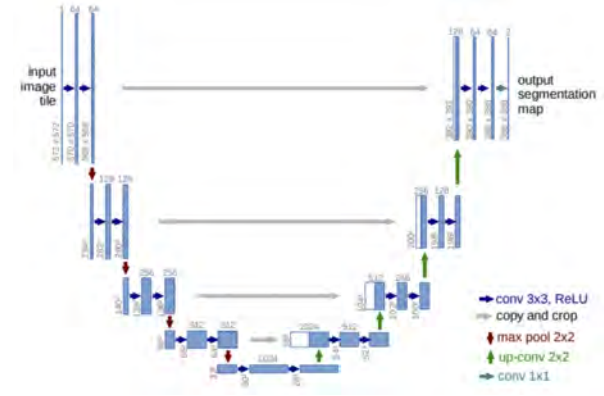
Figure 17: General architecture of UNET (Ronneberger et al. (2015))

similarly to the process of UNET. It separates 3D scans into patches, which are then linearly projected into token embeddings. Similar to Vision Transformers, the tokens are then handled by the self-attention block. The patch taken is huge (e.g. 16 x 16 x 16) to keep the complexity less as this will prevent the series length of input to be too lengthy. 18 As a result, Multi-dimensional CT
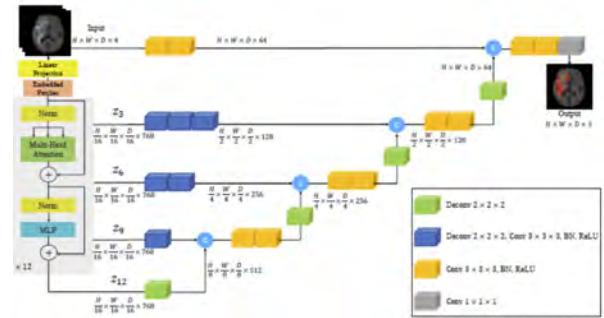
Figure 18: General architecture of UNETR (Hatamizadeh et al. (2022))

scan is given as input to this model. It projects it into a 1D sequence of non-overlapping patches. As we know transformers cannot work on data in more than one dimension. Here, we can also see skip connections joining encoder to the decoder. The positional values are encoded with the input and passed further in the pipeline.

3.3.3. UTNETV2

Gao et al. (2022) claim the best performance in medical image segmentation. It obtained state-of-the-art results. In their architecture, they made three contributions

- Depth-wise Separable Convolution to make it translation invariant. Depth-wise convolution works by dividing the convolution in two parts, i.e., Filtering stage and combination stage. in filtering stage the convolutions are applied to 1 channel per kernel and in combination stage the convolution is applied to the whole result of filtering stage

- Bi-directional Attention to reduce the complexity and compress large token maps to small semantic maps

- Multiscale features fusion of those small semantic maps. This is an important feature for us because in medical images sizes vary greatly.

This model was the good choice for us because it performs well with small dataset even if there are translations and size differences (as our source of images were different so highly likely)
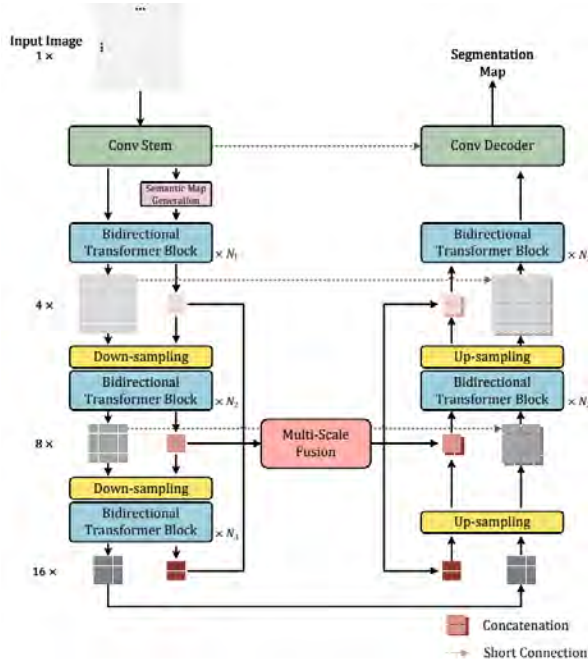
Figure 19: The complete architecture of UTNETV2 (Hatamizadeh et al. (2022))

UTNETV2, like other transformer based architectures, makes token embeddings of the image features by convolution block. These are then downsampled four times. Also, these features are given to the decoder side through skip connections. The Bi-directional attention unit reduces unnecessary tokens by projecting a compact semantic token map from the high-resolution token map as a semantic summary at every level using depthwise separable convolution.

Later in the decoder, we use upsampling layers and the info given by skip connections from the encoder part so it can combine the features and produce the segmentation result. (see figures 19,20).

Now after segmentation, we needed to find the distance, and for that we need to know the position of the electrodes from a real case scenario. This positions from the electrodes can be extracted through Nautilus.

3.3.4. Nautilus

Nautilus (Margeta et al.) provides a comprehensive collection of research tools for pre- and post-operative
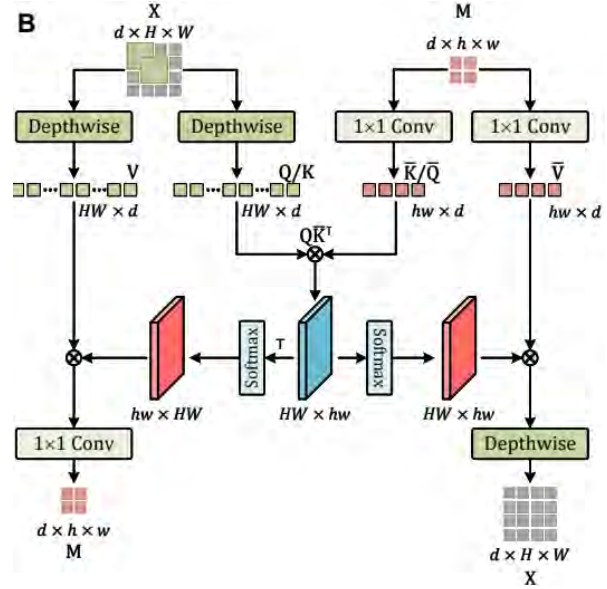
Figure 20: Modified version of Bi-directional multihead attention (Hatamizadeh et al. (2022))

CT scans. This examines cochlea automatically by image processing and also provides interactive visualization via a web browser for CI implantation (see fig 22, 21).
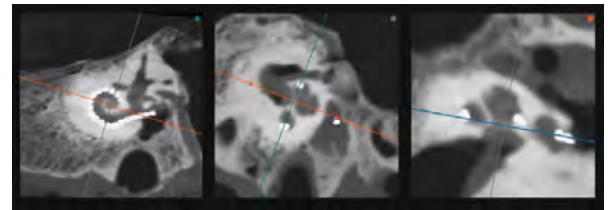
Figure 21: Electrodes insertion in same example but from different views in CT scan

Figure 22: Visual representation of electrodes and insertion in Nautilus

It segments the cochlea from pre-operative images and extracts electrode locations from a postoperative CT image using CNNs and geometrical inference before registering the information to compute metrics such as cochlear size and shape, characteristic frequencies at each contact, distance of electrodes to estimated basilar membrane position, and other metrics useful to surgeons, audiologists, and statisticians.

So here we can observe that the electrode placement with the segmentation model of cochlea.
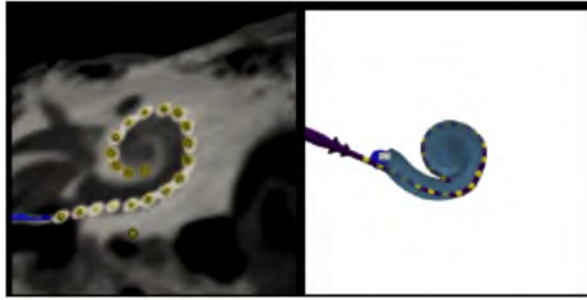
Figure 23: Left image is a post-operative CT scan showing white intensities as points of actual electrodes insertions and image shows the electrodes positions estimated within a 3D model of cochlea estimated from the segmentation

From this platform, we obtained the electrode placement coordinates of the test scans to compute their distance from the facial nerve. This distance will be used in finding out its significance in facial nerve stimulation (FNS).

4. Results

To compare all the above mentioned architectures, and evaluate the performance on our problem we used several metrics and plot the loss function evolution.

4.1. Loss functions

In deep learning, loss functions find out the difference between actual value and the predicted value. This is an optimization problem so it wants to minimize this value. In our experiments we used these loss functions. The value computed is used to adjust the weights during the backpropagation.

All the experiments below will be done on **UNET** with learning rate of 3e-4 and Adam optimizer (Kingma and Ba (2014)).

4.1.1. Cross-entropy loss function

It basically penalizes the wrong output by a very large number as it is the log of that number. So if we predict a probability of 0.2 while the actual value is 1 then it will give us penalization.

$$L_{cross-entropy} = \sum_{i=1}^{M} y_i \log(p_i)) \tag{1}$$

Where i ranges from 1 to total number of classes. In our case it was a good starting point but the performance was sub-optimal and only predicted the cochlea. The reason is that we have unbalanced data more in favor of Cochlea than Facial nerve. Here we can see a really good prediction of the cochlea but no signs of facial nerve. (see figure 24)
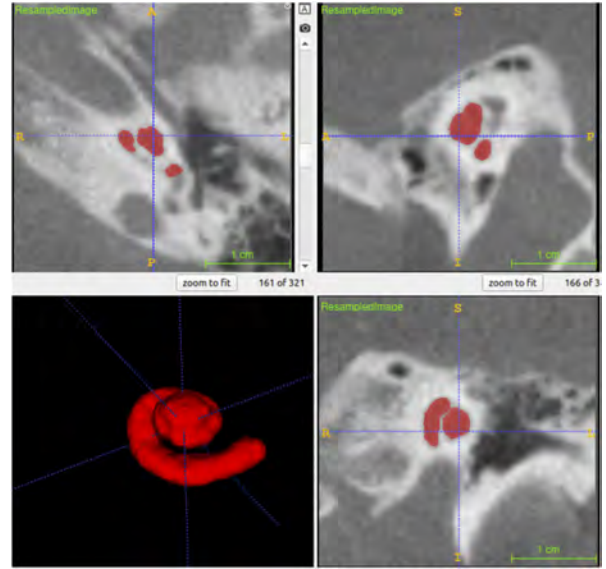
Figure 24: Prediction done on a validation image

4.1.2. Dice loss with cross-entropy(equal weights)

Cross-entropy is used to compute similarity between two images. However when we have the problem of class imbalance then it is much better to use Dice loss. Cross-entropy is a suboptimal loss function when the data data distribution is not balanced (Maier-Hein et al. (2022)) as is the case of fine structures we are dealing with. For experimentation we tried to use both the loss functions together. But even then cross-entropy made it more difficult for the model to learn the structure of facial nerve. So we had a similar prediction for our input.(see figure 25) Here we can see although the loss
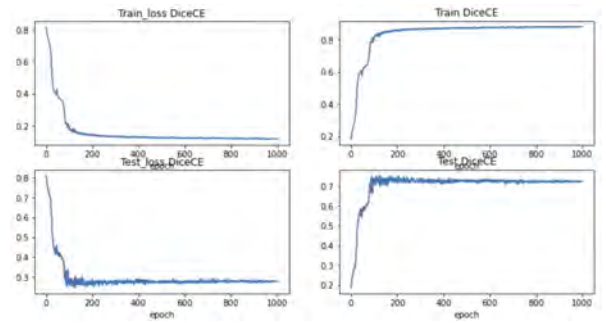
Figure 25: Loss curve and (1-loss) curve

is decreasing yet it is not giving us a good representation of how it is performing in reality. This achieved these numbers with just segmenting cochlea without facial nerve.

4.1.3. Dice loss

Now we evaluated the convergence of our model using only Dice loss (see figure 26). The formula for dice loss is shown where p_true is the actual probability for a voxel to belong to a class and p_{pred} is the predicted

probability of the respective voxel.

$$L_{dice} = \frac{2*\sum(p_{true}*p_{pred})}{\sum p_{true}^2 + \sum p_{pred}^2} \qquad (2)$$

We can see that the loss on the test data is similarly decreasing so that means we are not overfitting to the training set much. But this experiment seems rather biased with one dominant class.
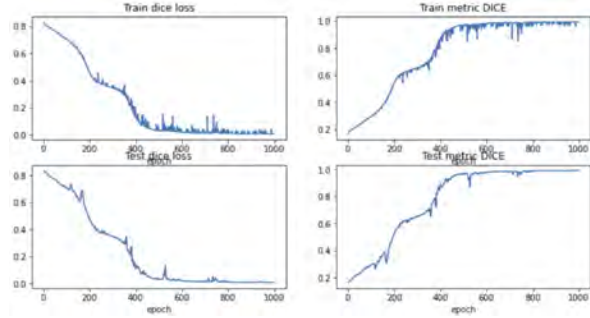
Figure 26: Loss curves and (1-loss) curve

Here the prediction was good for the train set (see figure 27) but in the test set(see figure 28,29), it was mixing cochlea label with the facial nerve.
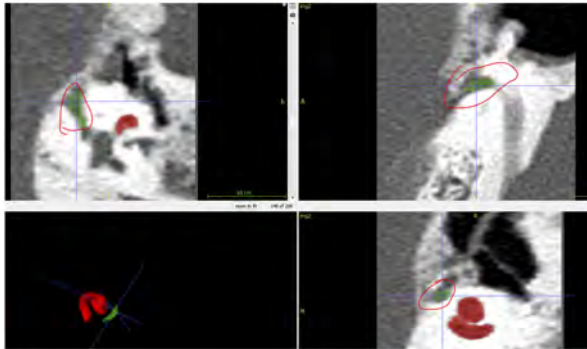
Figure 27: Segmentation result with Dice Loss of training set
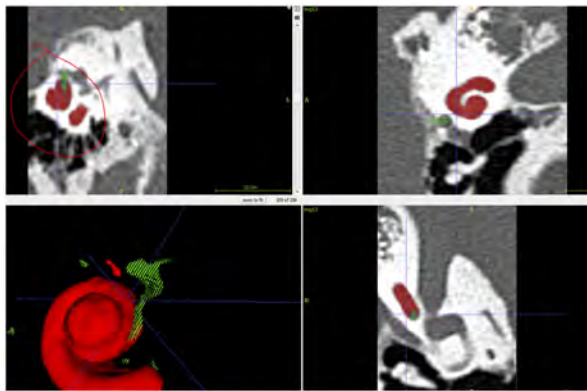
Figure 28: Segmentation result with Dice Loss of testing set

Here, we even checked the predicted probability map and evaluated different thresholds but still it was con-
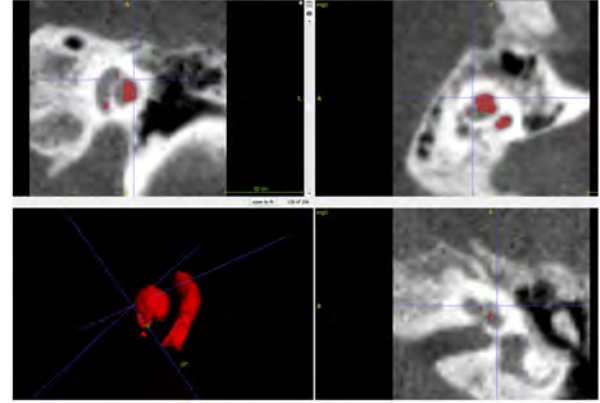
Figure 29: Another example of segmentation result with Dice Loss in testing set. We see that cochlea segmentation region is split into two which is not expected.

cluded by us that the model is confused about the structure of the facial nerve.
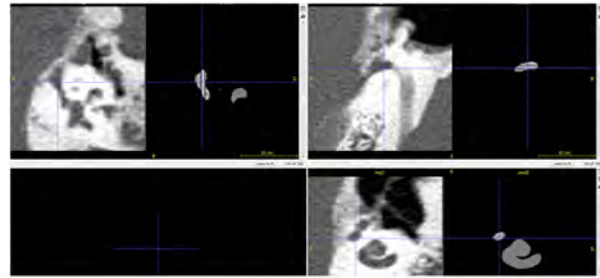
Figure 30: Probability map of the prediction with the scan on left side

(see figure 30) Here we can see that even changing the threshold does not change the fact that the prediction is not accurate. (see figure 31) This problem was on all the slices so it was not independent of any axis or slices. Also, showing there is no problem in the labeling of scans

4.1.4. Generalized Dice Loss

Generalized Dice loss (Maier-Hein et al. (2022)) gives weights to each class according to the label frequencies. So, we evaluated this loss function that is designed to work even better incase of an unbalanced dataset which we imagine is still the case. Using all the same settings we got (see figure 32,33)

Here the predictions were much more accurate but it was giving us the wrong boundary to the facial nerve because it is very near the cochlea. This problem is important for us because we are looking to calculate the distance for the facial nerve stimulation so boundaries are very important for us.

4.1.5. DiceFocal Loss

Focal loss is a modified version of cross entropy loss as it also handles class imbalances with another parameter which gives more weights to hard examples.
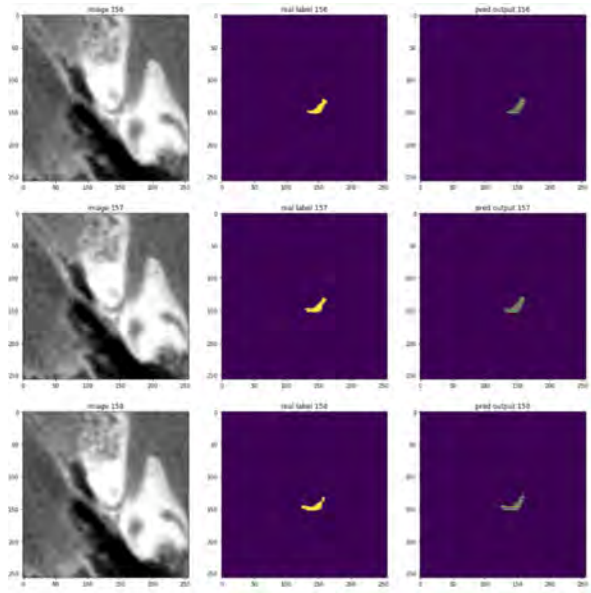
Figure 31: Slice by slice of single scan along with its real labels and predictions
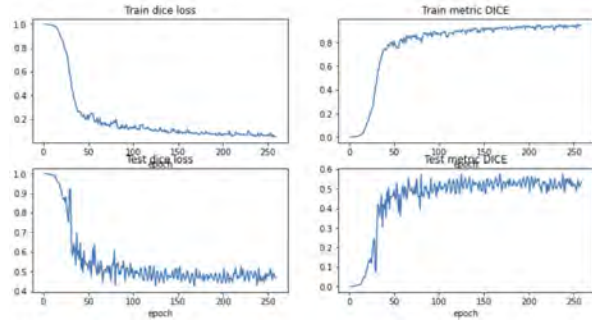
Figure 32: Loss curves and accuracy curves
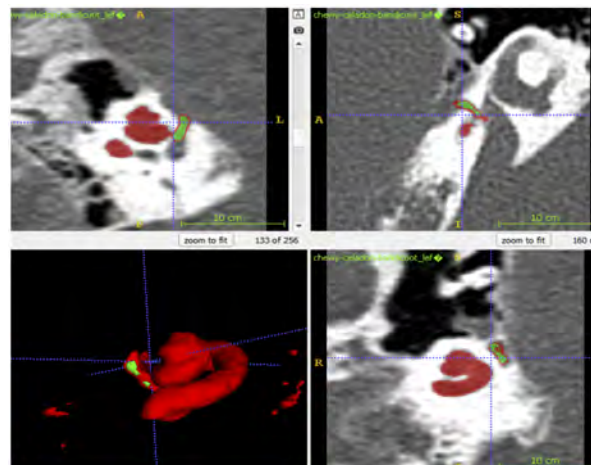
Figure 33: Prediction results on test image. Here we see that FN segmentation gets covered with cochlea label so possible solution can be to have more diverse training data or more augmentations to better cover the FN appearance variations.

This means to compute Dice loss and focal loss and returns the equally weighted sum of both(Diaz-Pinto et al. (2022)). Here we can change the weights of each loss.

$$L_{focal} = -(1 - p_i)^{\gamma} log(p_i) \tag{3}$$

Where p_i is the probability of a voxel to belong to a class. And γ is a hyperparameter that decides how much weights to give to a minority class.

$$L_{Dicefocal} = 0.5 * L_{focal} + 0.5 * L_{dice} \tag{4}$$

This loss performed the best especially for the important boundaries. This is what we will use with different architectures. (see figure 34)
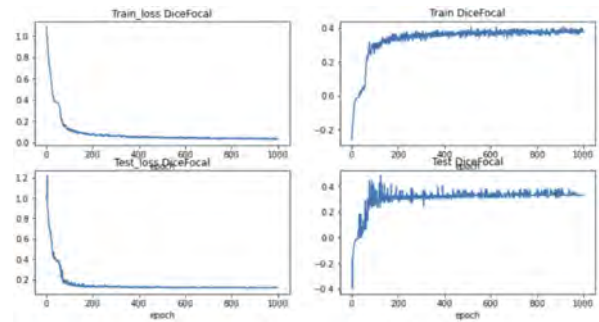
Figure 34: Loss curves and accuracy curves

4.2. Dice-Focal loss with UNETR

Here we tried different parameters because of demanding hardware resources required because of the huge architecture. We tried changing parameters but found out that the best results were noticed when using default parameters. (see table 2)

Parameter	Exp 1	Exp 2
imgsize	224	256
hiddensize	768	512
numheads	12	8
featuresize	16	8
mlpdim	3072	1536
posembed	'conv'	'conv'
convblock	True	True
resblock	True	True
normname	'instance'	'instance'

Table 2: Overview of parameter values for UNETR.

Although many more experiments can be done to achieve the best results. (see figure 35,36)
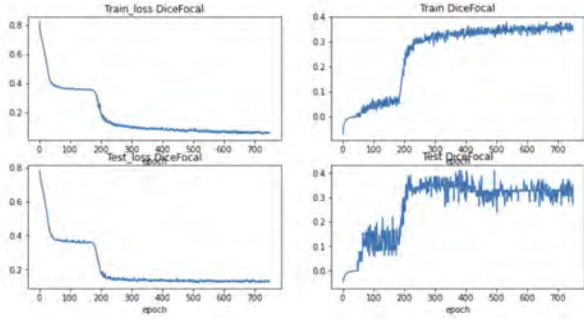
Figure 35: Loss curves and accuracy curves with experiment 1 parameters
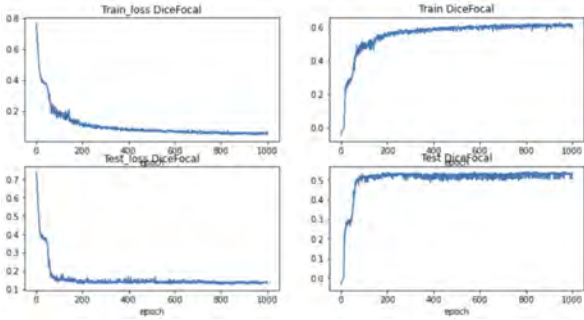
Figure 36: Loss curves and accuracy curves with experiment 2 parameters

4.3. Dice-Focal loss with UTNETV2

Here we have a lot of parameters to optimize but due to hardware limitations and time constraints we went with the parameters used by the original authors in their research paper. (see figure 37)
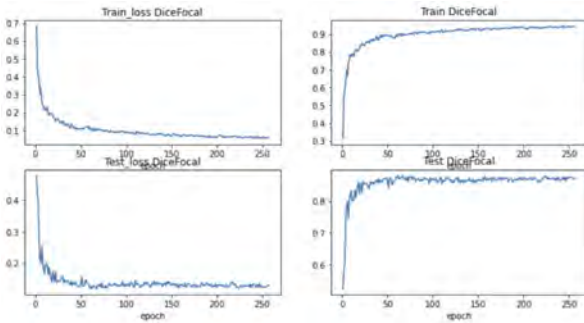
Figure 37: Loss curves and accuracy curves of UTNETV2

4.4. Evaluation

For evaluation we used three metrics which are as follows:

- Dice Coefficient (F1 Score): (Shamir et al. (2019)) explains the Dice coefficient. Basically, two times the intersecting area divided by the total number of pixels in both scans is the Dice Coefficient.

- Hausdorff distance: (Dubuisson and Jain (1994)) defines the metric as the distance of the maximum difference between two distinct points between 3D segmentation prediction and our 3D annotated label

- Volumetric difference: It is the surface distance, computed as absolute value of real segmentation minus predicted, divided by total.

4.4.1. Quantitaive Results

Here we represent our results on Validation and Test data. (Figures 38,39,40,41,42,43)

	count	mean	std	min	50%	95%	max
dsc_bk	15.0	0.999171	0.000522	0.997970	0.999377	0.999655	0.999664
dsc_C	15.0	0.742362	0.271337	0.000484	0.838454	0.930987	0.931637
dsc_FN	15.0	0.593613	0.200432	0.000000	0.662046	0.763024	0.828470
hd_bk	15.0	2.099144	1.639554	0.707107	1.276715	4.667904	5.981639
hd_C	15.0	1.951170	1.735842	0.519615	0.969536	4.667904	5.981639
hd_FN	15.0	1.322314	1.403116	0.458258	0.989949	3.231957	6.165225
vd_bk	15.0	0.001342	0.001169	0.000108	0.000913	0.003795	0.004069
vd_C	15.0	0.332521	0.276142	0.043607	0.265881	0.904288	0.999758
vd_FN	15.0	0.490275	0.494009	0.112253	0.342710	1.163343	2.115513

Figure 38: UNET Results on Validation data. Dice coefficient is good for both cochlea and FN but HD is high for both structures.

	count	mean	std	min	50%	95%	max
dsc_bk	15.0	0.999617	0.000119	0.999353	0.999671	0.999734	0.999748
dsc_C	15.0	0.927400	0.035344	0.837925	0.941358	0.960132	0.962277
dsc_FN	15.0	0.594251	0.202870	0.000000	0.677274	0.781377	0.797001
hd_bk	15.0	0.986141	0.508701	0.565685	0.806226	1.891406	2.491987
hd_C	15.0	0.670535	0.392406	0.300000	0.519615	1.378331	1.794436
hd_FN	15.0	1.632607	3.180958	0.565685	0.700000	5.150357	13.072873
vd_bk	15.0	0.000302	0.000290	0.000032	0.000181	0.000770	0.001113
vd_C	15.0	0.066476	0.066545	0.003694	0.048403	0.185952	0.224550
vd_FN	15.0	0.295958	0.279231	0.000945	0.226277	0.856719	0.986349

Figure 39: UNETR Results on Validation data. Dice coefficient is good for both cochlea and FN. HD is low for both structures.

	count	mean	std	min	50%	95%	max
dsc_bk	15.0	0.999562	0.000302	0.998519	0.999632	0.999733	0.999736
dsc_C	15.0	0.903984	0.106202	0.527091	0.930322	0.952203	0.953644
dsc_FN	15.0	0.612608	0.187864	0.000000	0.686328	0.758035	0.759614
hd_bk	15.0	1.141325	0.767214	0.574456	0.900000	2.680931	3.385262
hd_C	15.0	0.881680	0.872583	0.300000	0.574456	2.680931	3.385262
hd_FN	15.0	1.800133	3.734136	0.458258	0.787401	5.513414	15.264993
vd_bk	15.0	0.000433	0.000606	0.000003	0.000291	0.001138	0.002550
vd_C	15.0	0.101662	0.152935	0.004881	0.072054	0.272373	0.635660
vd_FN	15.0	0.223849	0.208615	0.015976	0.175117	0.589551	0.799577

Figure 40: UTNETV2 Results on Validation data. Dice coefficient is good for both cochlea and FN. HD is low for both structures but it is higher than that of UNETR.

	count	mean	std	min	50%	95%	max
dsc_bk	13.0	0.999196	0.000553	0.997658	0.999345	0.999599	0.999600
dsc_C	13.0	0.792236	0.239123	0.040840	0.878089	0.922533	0.927658
dsc_FN	13.0	0.585688	0.196728	0.000000	0.624531	0.743107	0.746176
hd_bk	13.0	1.662803	1.304036	0.547723	1.135782	3.904217	5.369358
hd_C	13.0	1.594256	1.348101	0.538516	0.979796	3.904217	5.369358
hd_FN	13.0	1.288045	1.588559	0.538516	0.830662	3.348331	6.521503
vd_bk	13.0	0.001221	0.001264	0.000209	0.000579	0.003563	0.004694
vd_C	13.0	0.270796	0.249141	0.058729	0.191138	0.709608	0.979155
vd_FN	13.0	0.288455	0.230154	0.004749	0.256798	0.649583	0.875809

Figure 41: UNET Results on Test Data. Dice coefficient is good for both cochlea and FN but HD is high for both structures.

	count	mean	std	min	50%	95%	max
dsc_bk	13.0	0.999564	0.000155	0.999137	0.999607	0.999719	0.999756
dsc_C	13.0	0.922391	0.038428	0.803902	0.930359	0.947667	0.949315
dsc_FN	13.0	0.610597	0.199738	0.156093	0.679279	0.782184	0.824009
hd_bk	13.0	0.960931	0.412913	0.489898	0.836660	1.675703	2.100000
hd_C	13.0	0.831861	0.410700	0.447214	0.707107	1.598855	1.907878
hd_FN	13.0	0.775992	0.456607	0.424264	0.640312	1.562872	2.126029
vd_bk	13.0	0.000404	0.000443	0.000092	0.000186	0.001156	0.001688
vd_C	13.0	0.080245	0.084390	0.008511	0.068805	0.226549	0.324496
vd_FN	13.0	0.267193	0.228699	0.007018	0.234223	0.609571	0.858504

Figure 42: UNETR Results on Test data. Dice coefficient is good for both cochlea and FN. HD is low for both structures.

	count	mean	std	min	50%	95%	max
dsc_bk	13.0	0.999449	0.000435	0.998045	0.999583	0.999679	0.999697
dsc_C	13.0	0.883605	0.143361	0.410889	0.921815	0.945154	0.951434
dsc_FN	13.0	0.555120	0.244287	0.000000	0.656037	0.762111	0.781040
hd_bk	13.0	1.357287	1.406267	0.678233	0.943398	3.378262	5.958188
hd_C	13.0	1.007340	0.994504	0.360555	0.748331	2.631569	4.091455
hd_FN	13.0	1.992443	4.178439	0.583095	0.830662	7.059303	15.886157
vd_bk	13.0	0.000635	0.000862	0.000152	0.000357	0.001919	0.003425
vd_C	13.0	0.139261	0.182622	0.039768	0.084792	0.393271	0.735766
vd_FN	13.0	0.163408	0.169706	0.008345	0.116279	0.452832	0.612238

Figure 43: UTNETV2 Results on Test data. Dice coefficient is good for both cochlea and FN. HD is low for both structures but it is higher than that of UNETR.

4.4.1.1 Comparisons of cochlea segmentation

Here are all the research papers results published till now on cochlea segmentation. (see table 3). These results are discussed in detail in section 5.

		Average Symmetric Surface Distance ST (mm)	MaxSurface Dist. error ST(mm)	Mean Surface dist.error undiff. cochlea	MaxSurface dist. error undiff. cochlea	Cochlea - DICE
Ours	2022	-	-	-	-	0.95
Nautilus	2022/02/10	-	-	-	-	0.86
Wang et al. (2021)	2021	0.22	0.50	-	-	-
Banalagay et al. (2021)et al.	2021	0.11	0.87	-	-	-
Lv et al. (2021)et al	2021	-	-	-	0.25	0.9
Hussain et al. (2021)	2021	-	-	-	-	0.9
Nikan et al. (2020)et al.	2021	-	-	0.27	-	-
Neves et al. (2021)	2021	-	-	-	-	0.91
Heutink et al. (2020)et al.	2020	-	-	-	-	0.9
Zhang et al. (2019)	2019	0.08	-	-	-	-
Ruiz Pujadas et al. (2018)et al.	2018	-	-	0.11	0.58	-
Demarcy (2017)et al.	2017	0.12	0.92	-	-	-
Gerber et al. (2017)et al.	2017	-	-	-	-	0.88
Kjer et al. (2015)et al.	2015	-	-	0.22	-	-
Noble et al. (2011)	2011	0.21	0.8	-	-	-
Abeysinghe et al. (2008)	2008	-	-	-	-	0.72

Table 3: Overview of cochlea techniques and their results.

4.5. Postprocessing

The segmentation results were generally satisfactory. But in some of the cases there needed to be improvements as we know in CT scans, the contrast values are very similar so there were artifacts included in predictions.

4.5.1. Connected Component

For removing these artifacts, we just had to keep the largest area structure for each label. For the task we employed the approach of connected components. (see figure 44)
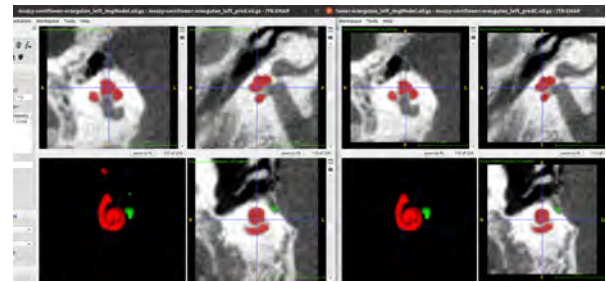
Figure 44: Example of output on left and post-processed output on right. Red the cochlear segmentation, in green the FN segmentation

4.5.2. Distance calculations

Our main objective was to find how close the electrodes are from the FN. So, we used the same origin and direction as the original scan while obtaining our segmentation. We then calculate the Maurer distance map on this segmented output. The electrode distances are extracted from our web-based solution, Nautilus. We extracted the electrode positions of the test scans to compute the distances in them.

4.5.3. Maurer Distance

Here we take one label as a base and produce a distance map for that specific label. In our case, we took the label of facial nerve. This map is zero on the boundaries and inside the more you go away from the boundary, it becomes negative and outside the boundary the more you go away the larger the distance. (see figure 45)
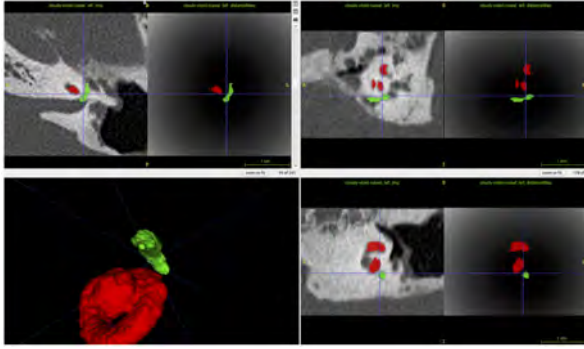
Figure 45: An example of the test CT scan with both CT scan and its distance map on left and right and our models prediction overlaid on top for better understanding.

Using this map, we can find out the distance between electrodes to the FN by just looking at the intensity value at those coordinates. In calculating this map, we made sure that the header content was accurately mapped to the header in Nautilus so the coordinates correspond correctly. (see figure 46)

We also performed manual checking with ITK-SNAP and it corresponded with our measurements. (see figure 47)
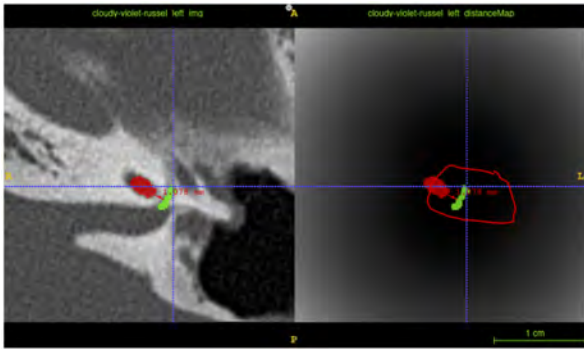
Figure 46: An example of the test CT scan with both CT scan and its distance map on left and right and our models prediction overlaid on top for better understanding.

5. Discussion

We can see from our results that our method seems to work much better than previous studies (up to five percent or 5% on the test set) (of our best model).

CordX	CordY	CordZ	ElectrodeDistance
174.968794555162	205.244052836325	126.376873666323	32.049991607666
165.312257355426	198.401881266226	128.725785930276	28.8499965667725
155.501442754446	192.432323975864	127.166031424904	31.2500114440918
145.477042554882	185.366556997307	125.721808869791	36.8500099182129
135.078138904185	179.533980900169	126.5521736316	41.8000068664551
126.121072203527	173.800670202614	132.046765027312	43.2399978637695
121.244299971548	167.767731815189	140.1376549156	41.9300003051758
119.506874847584	163.834420474857	150.27100571419	37.8899955749512
122.926841265225	159.154893671812	160.477805624603	32.0000038146973
127.768956413197	160.042572717221	171.183576626838	23.5500068664551
133.718179774073	163.659678422571	180.625566645262	16.4099998474121
142.811342511457	169.030638683302	185.910771385871	8.84000778198242
153.570101813766	173.826007006201	188.035512645709	3.96000075340271
163.047093048087	178.87104679492	184.622365260153	0.849996566772461
170.891080570101	184.102185909426	177.70121791779	0.200000002980232
175.666816307022	186.834153792537	167.674155572124	1.68999803066254
176.653076003736	185.968458697178	156.092495215819	5.7599983215332
172.836876188615	182.235349139966	146.023310873638	11.6399965286255
164.740713967157	177.105627031455	139.690303724407	18.2999935150146
155.076522950819	170.770022378024	137.800688079606	24.0599994659424

Figure 47: Final output of our pipeline. CSV file shows the electrodes coordinates along with their distances from facial nerve

Choosing the correct loss function was critical for obtaining our results. In general, loss functions are only used to optimize the model but they are not a good way to compare different architectures. Our experiments proved that dice loss will work for us but noticed that Dice in addition with Focal loss, with equal weights, worked even better. Dice loss works on class imbalance problem between easy and difficult examples but overlooks the imbalance between hard and easy scans. And Focal loss is a better version of Crossentropy loss that handles class imbalance by assigning more weights to hard or easily misclassified examples.

Before performing experiments, we assumed that UT-NETV2 will perform better than other networks because according to literature it performs well with small datasets even if there are translations and size differences (which was indeed our case). Also, in their paper, they were able to get better results than UNETR. But we ended up getting better Dice scores with UNETR which could be possible due to hyperparameters being used. The parameters used for UNETR have been tested and validated by MONAI (Diaz-Pinto et al. (2022)) with different datasets whereas the hyperparameters we used for UTNETV2 were the ones from the research paper for specific dataset (different from ours). Hyper-parameter search was out of scope for this project. We believe we can obtain a better performance by optimizing the parameters.

In the table of results, for Facial Nerve, Dice score will not be much reliable as it is a small structure. The related segment is both the narrowest (< 0.7 mm diameter) and shortest (3–5 mm length) segment of the FN (Gupta et al. (2013)). So we notice that the max Hausdorff distance to FN is very high for (UTNETV2). This is because in one of the scans it predicted FN to be far from original location but in others, it was satisfactory which is why the mean HD is 1.8-1.9 mm. This was because the resolution of that scan was very low before preprocessing so interpolation may have caused wrong prediction of FN. The other two architectures perform more consistently across all scans. Therefore

we should compare the architectures based on the 95% value. Overall, the UNETR performs better than the other two.

Regarding Cochlea, again the best results were with UNETR. If we observe closely to the values of Haus-forff distance of cochlea than it is 0.6mm and 0.8mm for validation and test set respectively which are sat-isfactory and shows that we can use these predictions. Similarly the same case with UTNETV2. But incase of UNET they are almost 2.0 mm which is not good con-sidering the whole structure of Cochlea is of 8-10 mm wide. So 2 mm means 20% error.

In this thesis, we worked on segmenting the part of facial nerve which is near to cochlea and then used that part of segmentation for our applications. After com-pleting the goal of this thesis we proceed to segment the facial nerve completely (also near the chorda tympani).

Chorda tympani originates from the mastoid segment of the facial nerve. Like the previous work, this part is really important during the surgery for CI. When the surgeon inserts the electrodes, he/she has to pass through a triangle-shaped V-like neural branching in or-der to enter cochlea through the round window without harming the nerves. This is the tricky part because there are very few voxels corresponding to the chorda tyam-pani and as in CT the contrast is a poor indicator for finding out nerves, it has not been done yet.

Currently, we have two more datasets one of CBCT and one of microCT both by Oticon Medical and some microCT scans from OpenEar dataset. Some additional scans of this dataset provided by Oticon Medical were again labeled by us by using the same strategies ex-plained in this paper. (see figure 48)
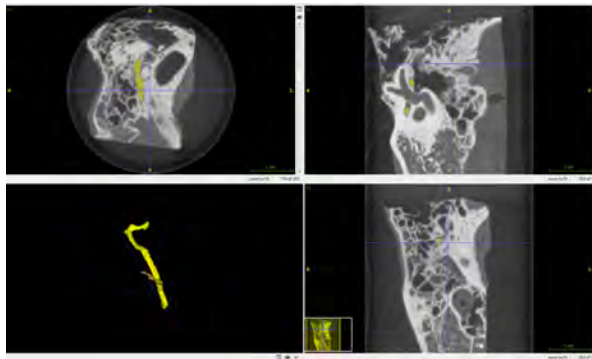
Figure 48: One example with labeled facial nerve and chorda tympani

Normal methods for segmentation were not working for this so we came up with an intelligent patch based approach which firstly identifies a region of interest in the CT scan and then divides it into patches. And now we can feed these patches for Unet based segmentation. (Figure 49,50)

For annotations, we used ITK-SNAP and did it man-ually for every scan after learning how radiologists see and figure our the structures in CT scans. There
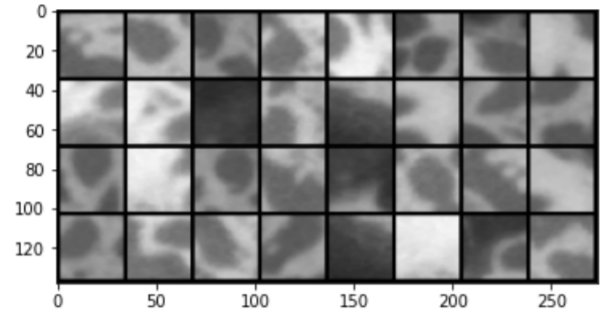
Figure 49: Patches extracted from CT example
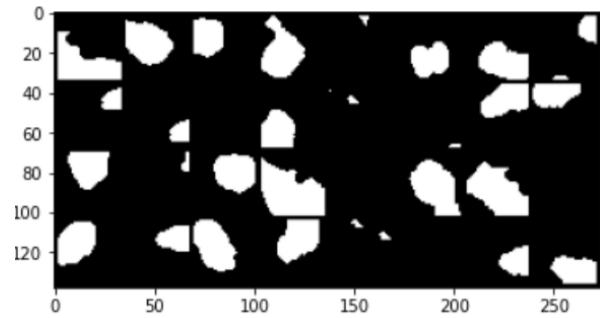
Figure 50: Corresponding masks of the patches extracted

is active-learning-based annotation tool MONAI Label (Diaz-Pinto et al. (2022)). It basically has segmenta-tion models that are always running and are pretrained on some medical imaging datasets. It works with 3D Slicer (Pieper et al. (2004)) tool with an active learn-ing MONAI label (Diaz-Pinto et al. (2022)) plugin. So this way we upload the scan on the network and also to Slicer and when we are labeling the scans we are also teaching the model about the anatomy of some nerve. So after few scans, models trained with MONAI Label can give some initial predictions which we can correct and the models continue to learn with more data. Using this tool can require fairly significant computational re-sources and there is a trade-off between using the given resources for exploration of new architectures and an-notating more data. In the future, we plan to use this tool to label more scans with Chorda Tympani.

Evaluating more segmentation models and using them in an ensemble using a voting or similar strategy could bring even further performance gains. Also, when there are changes in anatomy, such as between children and adults, the approach may not perform well (Reda et al. (2011)). A dataset containing paediatric scans will be needed to validate of the tool on non-adult scans.

6. Conclusions

The initial goal of this thesis was to create an auto-mated workflow to segment fine structures (facial nerve and cochlea) from the dataset of 3D CT scans. Despite

the fact that the soft tissue is notoriously difficult to segment in CT images, the created pipeline accurately characterizes both structures in the region of interest. The major objective is to deliver an end-to-end solution that will allow doctors to save time while still having access to important and trustworthy information for their diagnoses. This pipeline might help doctors and researchers study if and how the facial nerve stimulation is related to the distance to the cochlear implant within the cochlea.

The ability to segment the FN without requiring human intervention is a significant benefit, as this area is notorious for being difficult to characterize. We believe that this pipeline will enable doctors to determine its closeness to the cochlear structure in this vital location, allowing them to avoid FN stimulation caused by this proximity. This thesis also presents a method for automatically locating the FN in the region, where locating the FN might be problematic. Overall, we believe that the developed technology will be extremely useful to doctors to better understand the FNS and develop mitigation strategies. Using the feedback from doctors on the collected characterizations, and working on the points explained in the discussion, more breakthroughs might be made, and further improvements might be achieved.

Acknowledgments

References

Abeysinghe, S.S., Baker, M., Chiu, W., Ju, T., 2008. Segmentation-free skeletonization of grayscale volumes for shape understanding, in: 2008 IEEE International Conference on Shape Modeling and Applications, IEEE. pp. 63–71.

Baker, G.R., 2008. Tracking, modelling and registration of anatomical objects: the human cochlea. Ph.D. thesis.

Banalagay, R., Labadie, R.F., Noble, J., 2021. Validation of active shape model techniques for intra-cochlear anatomy segmentation in ct images, in: Medical Imaging 2021: Image Processing, International Society for Optics and Photonics. p. 115961M.

Beek, E., Pameijer, F., 2020. Head/neck: Temporal bone index, radiology department of the university medical centre of utrecht and the rijnland hospital. https://radiologyassistant.nl/head-neck/temporalbone/anatomy-2-0.

Campbell, A., 2020. Ear anatomy - english labels. https://www.campbellmedicalillustration.com/ear-anatomy-medical-illustrations.

de Castro, D.C., Marrone, L.C., 2021. Neuroanatomy, geniculate ganglion, in: StatPearls [Internet]. StatPearls Publishing.

Demarcy, T., 2017. Segmentation and study of anatomical variability of the cochlea from medical images. Ph.D. thesis. Université Côte d'Azur.

Diaz-Pinto, A., Alle, S., Ihsani, A., Asad, M., Nath, V., Pérez-García, F., Mehta, P., Li, W., Roth, H.R., Vercauteren, T., et al., 2022. Monai label: A framework for ai-assisted interactive labeling of 3d medical images. arXiv preprint arXiv:2203.12362 .

Dubuisson, M.P., Jain, A.K., 1994. A modified hausdorff distance for object matching, in: Proceedings of 12th international conference on pattern recognition, IEEE. pp. 566–568.

Elliott, S.J., Shera, C.A., 2012. The cochlea as a smart structure. Smart Materials and Structures 21, 064001.

Fang, C.H., Chung, S.Y., Mady, L.J., Raia, N., Lee, H.J., Ying, Y.L.M., Jyung, R.W., 2017. Facial nerve stimulation outcomes after cochlear implantation with cochlear-facial dehiscence. Otolaryngology Case Reports 3, 12–14.

Fauser, J., Stenin, I., Bauer, M., Hsu, W.H., Kristin, J., Klenzner, T., Schipper, J., Mukhopadhyay, A., 2019. Toward an automatic preoperative pipeline for image-guided temporal bone surgery. International Journal of Computer Assisted Radiology and Surgery 14, 967–976.

Gao, Y., Zhou, M., Liu, D., Metaxas, D., 2022. A multi-scale transformer for medical image segmentation: Architectures, model efficiency, and benchmarks. arXiv preprint arXiv:2203.00131 .

Gare, B.M., Hudson, T., Rohani, S.A., Allen, D.G., Agrawal, S.K., Ladak, H.M., 2020. Multi-atlas segmentation of the facial nerve from clinical ct for virtual reality simulators. International Journal of Computer Assisted Radiology and Surgery 15, 259–267.

Gerber, N., Reyes, M., Barazzetti, L., Kjer, H.M., Vera, S., Stauber, M., Mistrik, P., Ceresa, M., Mangado, N., Wimmer, W., et al., 2017. A multiscale imaging and modelling dataset of the human inner ear. Scientific data 4, 1–12.

Gupta, S., Mends, F., Hagiwara, M., Fatterpekar, G., Roehm, P.C., 2013. Imaging the facial nerve: a contemporary review. Radiology research and practice 2013.

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H.R., Xu, D., 2022. Unetr: Transformers for 3d medical image segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574–584.

Hatch, J.L., Rizk, H.G., Moore, M.W., Camposeo, E.E., Nguyen, S.A., Lambert, P.R., Meyer, T.A., McRackan, T.R., 2017. Can preoperative ct scans be used to predict facial nerve stimulation following ci? Otology & neurotology: official publication of the American Otological Society, American Neurotology Society [and] European Academy of Otology and Neurotology 38, 1112.

Heutink, F., Koch, V., Verbist, B., van der Woude, W.J., Mylanus, E., Huinck, W., Sechopoulos, I., Caballo, M., 2020. Multi-scale deep learning framework for cochlea localization, segmentation and analysis on clinical ultra-high-resolution ct images. Computer Methods and Programs in Biomedicine 191, 105387.

Hussain, R., Lalande, A., Girum, K.B., Guigou, C., Bozorg Grayeli, A., 2021. Automatic segmentation of inner ear on ct-scan using auto-context convolutional neural network. Scientific Reports 11, 1–10.

Johnson, H.J., McCormick, M., Ibáñez, L., Consortium, T.I.S., 2013. The ITK Software Guide. third ed. Kitware, Inc. *In press*.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Kjer, H.M., Vera, S., Fagertun, J., Gil, D., González-Ballester, M.Á., Paulsen, R., 2015. Image registration of cochlear μct data using heat distribution similarity, in: Scandinavian Conference on Image Analysis, Springer. pp. 234–245.

López Diez, P., Sundgaard, J.V., Patou, F., Margeta, J., Paulsen, R.R., 2021. Facial and cochlear nerves characterization using deep reinforcement learning for landmark detection, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 519–528.

Lv, Y., Ke, J., Xu, Y., Shen, Y., Wang, J., Wang, J., 2021. Auto-

matic segmentation of temporal bone structures from clinical conventional ct using a cnn approach. The International Journal of Medical Robotics and Computer Assisted Surgery 17, e2229.

Maier-Hein, L., Reinke, A., Christodoulou, E., Glocker, B., Godau, P., Isensee, F., Kleesiek, J., Kozubek, M., Reyes, M., Riegler, M.A., et al., 2022. Metrics reloaded: Pitfalls and recommendations for image analysis validation. arXiv preprint arXiv:2206.01653 .

Margeta, J., Demarcy, T., Lopez Diez, P., Hussain, R., Vandersteen, C., Guevarra, N., Delingette, H., Gnansia, D., Kamaric Riis, S., Patou, F., . Nautilus: A clinical tool for the segmentation of intracochlear structures and related applications .

Neves, C., Tran, E., Kessler, I., Blevins, N., 2021. Fully automated preoperative segmentation of temporal bone structures from clinical ct scans. Scientific reports 11, 1–11.

Nikan, S., Van Osch, K., Bartling, M., Allen, D.G., Rohani, S.A., Connors, B., Agrawal, S.K., Ladak, H.M., 2020. Pwd-3dnet: A deep learning-based fully-automated segmentation of multiple structures on temporal bone ct scans. IEEE Transactions on Image Processing 30, 739–753.

Noble, J.H., Labadie, R.F., Majdani, O., Dawant, B.M., 2011. Automatic segmentation of intracochlear anatomy in conventional ct. IEEE Transactions on Biomedical Engineering 58, 2625–2632.

Noble, J.H., Warren, F.M., Labadie, R.F., Dawant, B.M., 2008. Automatic segmentation of the facial nerve and chorda tympani in ct images using spatially dependent feature values. Medical physics 35, 5375–5384.

Pieper, S., Halle, M., Kikinis, R., 2004. 3d slicer, in: 2004 2nd IEEE international symposium on biomedical imaging: nano to macro (IEEE Cat No. 04EX821), IEEE. pp. 632–635.

Polak, M., Ulubil, S.A., Hodges, A.V., Balkany, T.J., 2006. Revision Cochlear Implantation for Facial Nerve Stimulation in Otosclerosis. Archives of Otolaryngology–Head Neck Surgery 132, 398–404.

Powell, K.A., Liang, T., Hittle, B., Stredney, D., Kerwin, T., Wiet, G.J., 2017. Atlas-based segmentation of temporal bone anatomy. International journal of computer assisted radiology and surgery 12, 1937–1944.

Reda, F.A., Noble, J.H., Rivas, A., McRackan, T.R., Labadie, R.F., Dawant, B.M., 2011. Automatic segmentation of the facial nerve and chorda tympani in pediatric ct scans. Medical physics 38, 5590–5600.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Ruiz Pujadas, E., Piella, G., Kjer, H.M., González Ballester, M.A., 2018. Random walks with statistical shape prior for cochlea and inner ear segmentation in micro-ct images. Machine Vision and Applications 29, 405–414.

Shamir, R.R., Duchin, Y., Kim, J., Sapiro, G., Harel, N., 2019. Continuous dice coefficient: a method for evaluating probabilistic segmentations. arXiv preprint arXiv:1906.11031 .

Sieber, D., Erfurt, P., John, S., Santos, G.R.D., Schurzig, D., Sørensen, M.S., Lenarz, T., 2019. The openear library of 3d models of the human temporal bone based on computed tomography and micro-slicing. Scientific data 6, 1–9.

Smith, R.J., Bale Jr, J.F., White, K.R., 2005. Sensorineural hearing loss in children. The Lancet 365, 879–890.

Voormolen, E.H., van Stralen, M., Woerdeman, P.A., Pluim, J.P., Noordmans, H.J., Viergever, M.A., Regli, L., Berkelbach Van Der Sprenkel, J.W., 2012. Determination of a facial nerve safety zone for navigated temporal bone surgery. Operative Neurosurgery 70, ons50–ons60.

Wang, D., Li, M., Ben-Shlomo, N., Corrales, C.E., Cheng, Y., Zhang, T., Jayender, J., 2021. A novel dual-network architecture for mixed-supervised medical image segmentation. Computerized Medical Imaging and Graphics 89, 101841.

Yushkevich, P.A., Gao, Y., Gerig, G., 2016. Itk-snap: An interactive tool for semi-automatic segmentation of multi-modality biomedical images, in: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE. pp. 3342–3345.

Yushkevich, P.A., Piven, J., Cody Hazlett, H., Gimpel Smith, R., Ho, S., Gee, J.C., Gerig, G., 2006. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage 31, 1116–1128.

Zhang, D., Banalagay, R., Wang, J., Zhao, Y., Noble, J.H., Dawant, B.M., 2019. Two-level training of a 3d u-net for accurate segmentation of the intra-cochlear anatomy in head cts with limited ground truth training data, in: Medical Imaging 2019: Image Processing, SPIE. pp. 45–52.
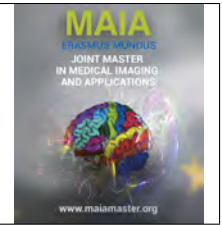
Spatio-Temporal Models to Evaluate the Critical View of Safety in Laparoscopic Cholecystectomy

Husam Nujaim[1], Adit Murali[1], Nicolas Padoy[1,2]

[1]*ICube, University of Strasbourg, CNRS, Strasbourg, France,* [2]*IHU Strasbourg, Institute of Image-Guided Surgery, Strasbourg, France*

Abstract

Laparoscopic cholecystectomy (LC), a minimally invasive surgery that aims to remove the gallbladder, is the most widely performed laparoscopic procedure today. The shift towards the minimally invasive procedure has, however, coincided with an increased rate of bile duct injury (BDI) which results in severe health and economical complications for the patients. The critical view of safety (CVS) approach has been shown to effectively prevent BDI; however, CVS is not always achieved for a variety of reasons. Recent works have begun to explore automated assessment of CVS from surgical video to drive increased CVS assessment rates. In this research, we propose two deep learning approaches to evaluate CVS, specifically focusing on spatio-temporal modeling. Our first approach incorporates temporal layers on top of the DeepCVS model, while our second approach foregoes the DeepCVS model, instead modeling a surgical video clip as a spatio-temporal region graph. This graph representation enables explicit modeling of individual anatomical structures and tools as well as their interactions over space and time, which in turn improves assessment of the CVS. Results show that both proposed models outperform the single frame DeepCVS baseline with mean average precision (mAP) = 60.92% and 59.55%, and balanced accuracy = 72.62% and 70.19% on the Endoscapes dataset for the spatio-temporal model and the spatio-temporal graph model respectively.

Keywords: Computer-Assisted Intervention, Laparoscopic Cholecystectomy, Critical View of Safety, Deep Learning, Spatio-Temporal Graph

1. Introduction

Computer-assisted intervention (CAI) is an emerging discipline that aims to improve the quality and precision of surgical procedures. Various building blocks for CAI have been developed, including models for surgical phase recognition (Kadkhodamohammadi et al. (2022), Cheng et al. (2022), Czempiel et al. (2021), Czempiel et al. (2020)), tool detection (Kondo (2021), Shimizu et al. (2021)), tool segmentation (Zhang et al. (2021), da Costa Rocha et al. (2019)), full scene semantic segmentation (Monasterio-Exposito et al. (2022), Alapatt et al. (2021)),and detection of anatomical structures (Owen et al., 2021). A natural next step is to develop interventional tools that can positively impact surgical safety.

A potential safety application of CAI is the prevention of bile duct injury (BDI) in laparoscopic cholecystectomy (LC), the most frequently performed laparo-scopic procedure today. While LC offers numerous advantages over open surgery which include decreased incisional pain, smaller open wounds or incisions, shorter hospitalizations, and faster recovery, it is associated with increased BDI rates. BDI in turn results in severe health Schreuder et al. (2020) and economical Halle-Smith et al. (2019) complications for patients, including longer recovery time, follow-up surgeries, degraded quality of life, and in some cases death. These complications additionally impose a significant economic burden to healthcare systems, to the tune of 1 billion dollars in the United States alone (Berci et al., 2013).

To tackle increasing BDI rates, Strasberg (1995) introduced the critical view of safety (CVS) approach, and later, Strasberg and Brunt (2010) introduced its rationale as the clear appearance of (1) cystic duct and cystic artery, (2) hepatocystic triangle, and (3) cystic plate. The CVS approach has become the standard for safe

LC wherein surgeons are instructed not to proceed with the resection of the cystic duct and artery before achieving the CVS criteria (Pucher et al., 2015),(Conrad et al., 2017). However, over the past decades, BDI rates have remained more or less stable Törnqvist et al. (2012), due to ineffective levels of CVS achievement.

Mascagni et al. (2020) and Mascagni et al. (2022) established a series of work exploring the feasibility and potential clinical value of automatic assessment of CVS in boosting CVS achievement rates and as a result, reducing potential BDIs. The latter introduced Deep-CVS, an artificial intelligence (AI) model to automatically identify CVS from endoscopic images utilizing the formalization of the CVS criteria defined by (Mascagni et al., 2020).

While DeepCVS illustrates the clinical feasibility of deep learning for CVS assessment, it is not quite comprehensive with regard to recent advancements in surgical video analysis; for instance, it does not include temporal information, which has become standard for various tasks in surgical video analysis (e.g. phase recognition) (see Sec. 2 for further limitations). The primary purpose of this work is to fill these gaps; to this end, we first begin by leveraging larger training and evaluation datasets and extending DeepCVS with several state-of-the-art temporal models. Then, observing that CVS criteria assessment is a fine-grained recognition task that relies on accurate identification of anatomical structures as well as their relationships in space and time, we investigate a novel approach using region graphs to model the surgical scene, using this region graph representation for downstream CVS assessment.

In summary, our contributions are as follows:

1. A spatio-temporal model to evaluate the critical view of safety in LC.
2. A novel multi-task framework for fine-grained spatio-temporal surgical video understanding using a Region Graph representation.
3. Improved validation of CVS prediction models by replacing a small hand-picked dataset of frames with a larger dataset of unsampled LC videos.

2. State of the art

2.1. Surgical Video Analysis

Surgical video analysis is crucial for intra-operative CAI systems and image-guided surgery. Recent works have focused on surgical phase recognition (Cheng et al. (2022), Czempiel et al. (2021), Gao et al. (2021), Guédon et al. (2021)), surgical tool recognition (Xue et al. (2022), Namazi et al. (2022), Liu et al. (2022), Alshirbaji et al. (2021)), surgical tool segmentation (Ni et al. (2022b), Yang et al. (2022), Sestini et al. (2022), Ni et al. (2022a), Zhao et al. (2022)), full scene semantic segmentation (Monasterio-Exposito et al. (2022), Alapatt et al. (2021)), detection of key anatomical structures

Owen et al. (2021), as well as instrument usage anticipation Yuan et al. (2021). Moreover, surgical action recognition (Nwoye et al. (2022b), Nwoye et al. (2020), Nwoye et al. (2022a)[1], Li et al. (2022)) has gained a lot of attention as it could help in modeling the interaction between surgical instruments and tissues at a fine-grained level which, in turn, could foster surgical monitoring systems and surgical safety Sharghi et al. (2020). The desire of improving surgical safety makes the critical view of safety (CVS) assessment an emerging task to be investigated.

2.2. Existing work in surgical safety

Even though the surgical safety is an emerging discipline in today's healthcare systems, very few methods have been proposed to tackle surgical safety challenges. The surgical safety applications require fine-grained analysis as they rely on the accurate recognition of anatomical structures. Recent work have focused on Go-No Go zones Madani et al. (2022), critical landmarks identification Tokuyasu et al. (2021), as well as the critical view of safety assessment (CVS) Mascagni et al. (2022). The latter developed the DeepCVS model which is designed to evaluate the criteria defining the CVS in laparoscopic cholecystectomy, yet they leveraged less amount of labeled data, and they designed their model for single frame predictions without considering the relationship among neighboring frames in the endoscopic videos. Furthermore, DeepCVS does not explicitly model fine-grained spatial and semantic relationships between anatomical structures, which is fundamental for accurate CVS assessment.

2.3. Existing work in graph-based methods

Scene graphs are semantically rich representations that model an image as a collection of objects/components and their semantic relationships. Prior work has focused on scene graph prediction Chen et al. (2019), image generation from scene graphs (Mittal et al. (2019), Johnson et al. (2018)), and scene graph generation Yang et al. (2018) among numerous other tasks. Recent works have also used graphs for spatio-temporal modeling (Wang and Gupta (2018), Khan and Cuzzolin (2021)), using them for downstream action recognition tasks. In the surgical domain, Islam et al. (2019), Seenivasan et al. (2022) investigate scene graph prediction in synthetic surgical video, but (1) do not explore the effectiveness of the predicted graphs for downstream tasks and (2) do not explore spatio-temporal graph approaches. In this work, we tackle these key limitations, extending the method of Wang and Gupta (2018) by using the region graph representation for CVS prediction rather than activity recognition.

---

[1] https://cholectriplet2021.grand-challenge.org/

3. Material and methods

3.1. Dataset

In this work, we use the Endoscapes dataset from Alapatt et al. (2021). Endoscapes is a dataset of 201 LC videos wherein one frame every 30 seconds (0.03 fps) is annotated with segmentation masks of anatomical structures and surgical tools (29 classes, 1,933 frames in total), and one frame in every 5 seconds is annotated independently by three experts with the three CVS criteria identified in Mascagni et al. (2022) (11090 frames in total). For the purposes of this work: (1) we define the ground-truth CVS annotation as the majority vote of the three annotators. Figure 1 shows the balanced accuracy between each annotator's assessments and the majority vote. (2) we utilize only 7 segmentation classes (including the background) that are associated with CVS prediction as introduced in (Alapatt et al., 2021).
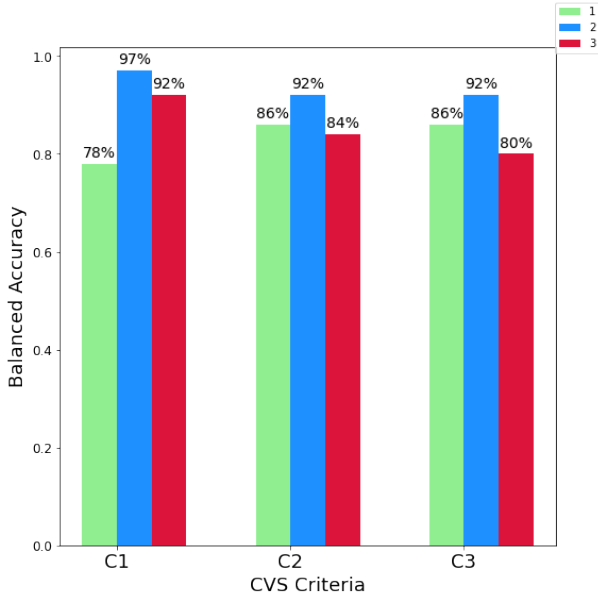
Figure 1: Balanced accuracy between each annotator with the derived majority annotation.

Dataset Split. We follow the splits used in Alapatt et al. (2021), separating the 201 videos into 120 training, 41 validation, and 40 test; Table 1 shows the resulting number of samples (frames) in each split.

Table 1: Number of samples in each split of the Endoscapes dataset.

Training Set	Validation Set	Test Set
6,960	2,331	1,763

Class Distribution. The dataset is characterized by significant class imbalance (see Fig. 2) with regard to CVS achievement for two reasons: (1) CVS is not achieved in all videos and (2) once the cystic duct and artery are clipped, which occurs soon after CVS achievement, CVS is no longer defined, and there are therefore few frames where CVS is fully achieved by nature. We describe our approaches to handle this class imbalance during training in Section 3.5.
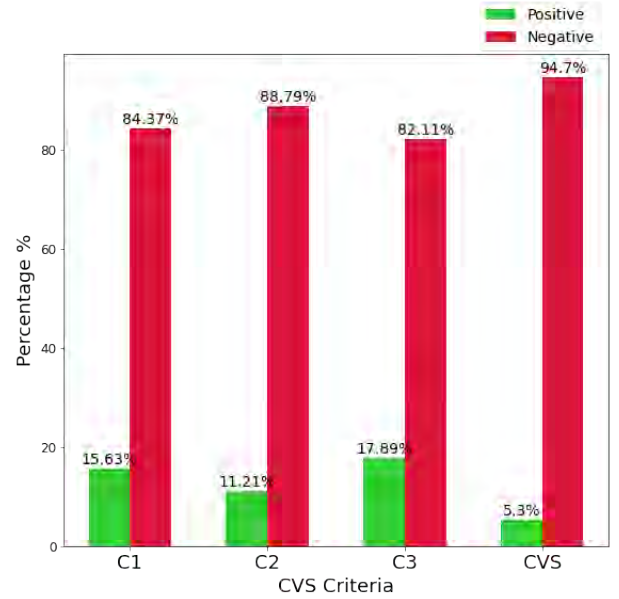
Figure 2: Class distribution (Training Set) for the 3 CVS criteria C1, C2, and C3. The last 2 right hand side bars (CVS) represent the class distribution when the 3 criteria are positive.

Temporal dataset. Since our dataset is sparsely annotated with CVS (only 1 labeled frame every 5 seconds), we assign the label of the frame at time t to the previous 4 frames ($t-1, t-2, .., t-4$). We refer to this dataset as the Temporal Dataset.

3.2. Single frame baseline (DeepCVS)

DeepCVS (Mascagni et al., 2022) is the current state-of-the-art model for predicting CVS achievement in LC video. This model was trained on a subset of the Endoscapes dataset containing 2854 images annnotated with CVS of which 402 were also annotated with segmentation masks. DeepCVS is designed for single frame predictions from endoscopic images.

DeepCVS architecture. DeepCVS is composed of two networks, a DeepLabV3-plus Chen et al. (2018) segmentation model and a shallow hand-designed CNN. The DeepLabV3-plus segmentation model is first trained to predict segmentation masks. Then this model is frozen and leveraged for CVS prediction as follows: the input image is first resized to $240 \times 427 \times 3$, representing height, width, and channels respectively, and passed through the segmentation network to obtain a $240 \times 427 \times 7$ output containing the segmentation probabilities of each of the 7 semantic classes. This predicted mask is concatenated with the original input image along the channel dimension to generate a $240, 427, 10$ input, which is finally forwarded to a CNN to predict CVS labels.
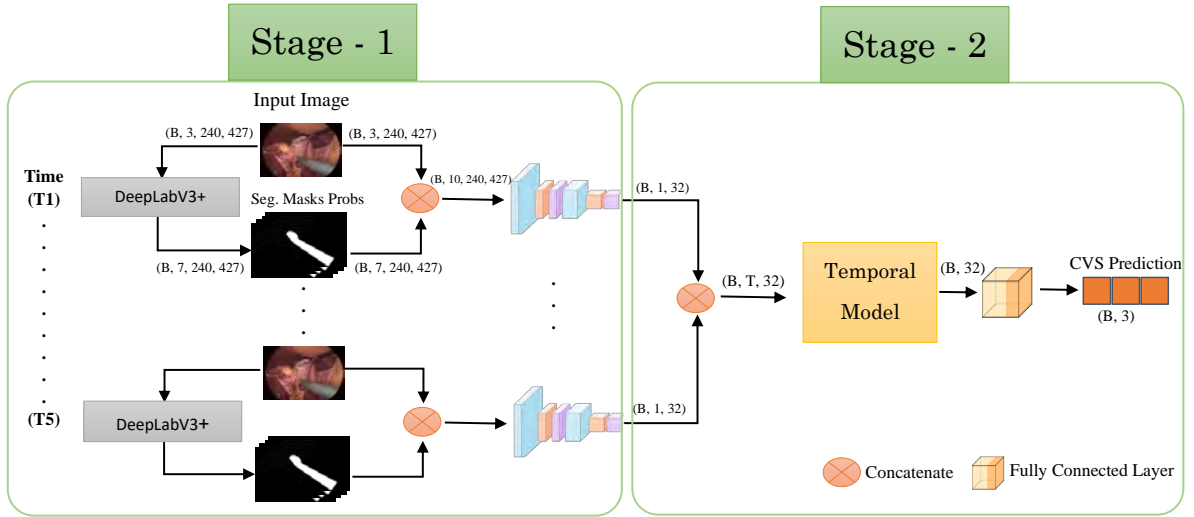
Figure 3: Spatio-Temporal DeepCVS Architecture

Reproducing DeepCVS. To enable fair comparisons, we retrain all components of DeepCVS using the complete Endoscapes dataset.

DeepCVS limitations. As DeepCVS is a single-frame model, it fails to leverage temporal information, which as noted by Mascagni et al. (2020), can be crucial for proper CVS assessment. In the following sections, we propose several approaches to tackle this critical limitation.

3.3. Two-Stage Spatio-Temporal Models

To incorporate temporal context for CVS prediction, we extend DeepCVS by replacing the final fully-connected (FC) layer with a temporal model, following several prior works in surgical workflow recognition. There are two approaches to train the resulting architecture:

- Single-Stage Training: Train the temporal Deep-CVS model end-to-end, backpropagating the gradient to the temporal layers as well as the shallow CNN.

- Multi-Stage Training: First train the DeepCVS model, freeze all weights, and only train the temporal layers, thus limiting gradient backpropagation to the temporal layers.

To limit computational complexity and enable longer temporal windows, we adopt the multi-stage training, additionally illustrated in Figure 3. For the temporal model, we investigate using LSTM (Twinanda et al., 2016), TCN (Czempiel et al., 2020), Transformer (Czempiel et al., 2021), RNN, and GRU. Importantly, we consider only causal models as the ultimate goal is real-time prediction.

We train the overall models as follows:

Stage 1. We first train the DeepCVS model to predict CVS labels using the annotated frames only. Then, we freeze the model's weights and remove the final fully connected (FC) layer from the DeepCVS to produce feature maps of shape $B \times C \times T \times H \times W$, where B, C, T, H, W denote batch size, number of channels, time, image height, and image width, respectively. Since the DeepCVS is a single frame model, the time dimension is always 1. We then forward these feature maps to an average-pooling layer, which averages the spatial features and produces $B \times 1 \times C$ features.

Stage 2. The $B \times 1 \times C$ features are computed for each frame in a clip, and then concatenated along the time dimension to produce a $B \times T \times C$ features. Finally, these features are forwarded to the aforementioned temporal models, which outputs spatio-temporal features of shape $B \times C$. These features are forwarded to a fully connected layer to predict 3 labels of shape $B \times 3$, corresponding to each CVS criterion, for the last frame of the clip.

3.3.1. Temporal Model Architecture Details

In this subsection, we explain the configuration details of each temporal layer. Note that we did not incorporate all the recurrent layers in a single model. Instead, we compared the performance of incorporating each recurrent layer with DeepCVS individually. A thorough comparison between the recurrent layers has been conducted by Chung et al. (2014).

Recurrent Neural Networks. RNNs (see Figure 4A) are a class of neural networks to model sequential data. We examine their capability to capture temporal dependencies for the task of CVS prediction. We use a single RNN layer which takes an input of shape $B \times T \times C$ and
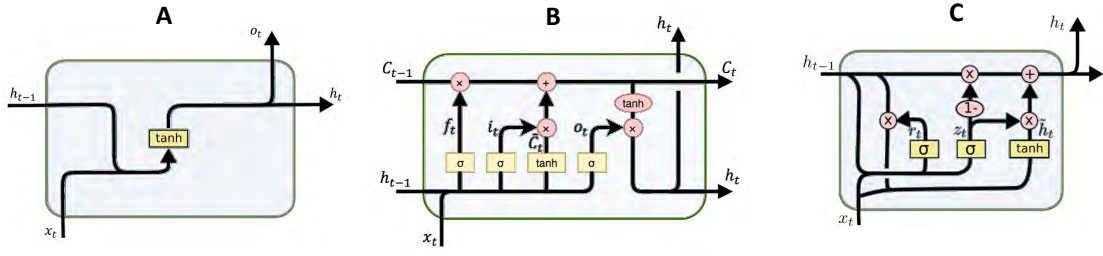
Figure 4: Recurrent Layers: A) RNN: where x_t is the input vector, h_t is a hidden layer vector, o_t is the output vector, and tanh is the activation function. B) LSTM: where h_t, C_t are hidden layer vectors, x_t is the input vector, and σ, tanh are activation functions. C) GRU: where h_t is the hidden layer vector, x_t is the input vector, and σ, tanh are activation functions.

then outputs features of shape $B \times C$ which contains the temporal information of all the T frames. We use 256 RNN units with tanh activation function. The kernel is initialized from a uniform distribution. The bias is initialized to zeros. We do not use any regularizers neither for the kernels nor for the bias.

It is well known that RNNs are limited in their ability to model long temporal sequences, due to phenomena including vanishing and exploding gradients. Therefore, we also investigate additional temporal models that can better handle these issues.

Long Short-Term Memory. The LSTM, shown in Figure 4.B, addresses the aforementioned limitations of RNNs by (1) incorporating a memory cell that maintains information for longer time and (2) incorporating gating layers (input and forget gates) which allow each LSTM layer to modulate the flow of temporal information, thus helping preserve long-range relationships. To initialize the LSTM, we use the same general configuration and initialization as with RNNs (256 LSTM units, tanh activation, etc.). We additionally use a Sigmoid activation function and include a bias term for the forget units.

Gated Recurrent Unit. The GRU, illustrated in Figure 4.C, is another temporal model that addresses the gradient flow issues of RNNs, but is faster and more parameter-efficient than the LSTM. It incorporates two gates: (1) The Update Gate which determines the amount of previous information that should be forwarded to the next state. This gate makes the GRU able to mitigate the problem of vanishing gradient. (2) The Reset Gate which controls the previous state either by keeping or eliminating the old information. We follow the same configuration as with the RNN and the LSTM.

Temporal Convolutional Networks. TCNs (Bai et al., 2018) have shown to perform better than LSTM/GRU specifically for modeling long-range sequences. Compared to recurrent layers, some advantages of TCNs are: (1) convolutions can be done in parallel, unlike RNNs, where predictions for later timesteps must wait for their predecessors to finish. This is because each layer uses the same filter. As a result, rather than processing a long input sequence sequentially as in RNN, a long input sequence can be handled as a whole

in TCNs for both training and evaluation. (2) TCNs, unlike recurrent layers, have a backpropagation path that is independent of the sequence's temporal direction. As a result, TCNs are robust to the exploding/vanishing gradients problems, which is a fundamental difficulty with RNNs that led to the development of LSTM. (3) Recurrent layers may consume a lot of memory storing partial results for their numerous cell gates, especially with long input sequences. In TCNs, however, the filters are shared across layers, and the backpropagation direction is solely determined by the network depth. As a result, TCNs utilize less memory than recurrent layers when dealing with long sequences.

We use a single TCN layer with kernel size = 3, number of filters in the convolutional layers = 64, and a list of the dilations = (1, 2, 4, 8, 16, 32), Moreover, we use ReLU activation function, a single residual block with skip connection from the input to the residual block, and causal padding in the convolutional layers.

Transformers. Transformers (Vaswani et al. (2017)) have shown to be effective not only in natural language processing, but also in computer vision problems (Khan et al., 2021). The self-attention mechanism is utilized by the transformer encoder and decoder. The fundamental idea of this mechanism is built on the assumption that not all the model's input data contain significant features based on which the model can make a prediction. Instead, the model should pay more attention to the relevant input features which help in making accurate predictions while paying less attention to other irrelevant features. This is similar to the intuition of Max-Pooling layer.

The Transformer encoder block is formed by multi-head attention layers which are an extra tweak to the self-attention mechanism (Tunstall et al., 2022). The "multi-head" name refers to factoring the output space of the self-attention layer into independent sub-spaces that are learned independently. Each subspace is called a "head". Three independent sets of dense projections are used to process the initial query, key, and value which results in three separate vectors. Additionally, neural attention is used to process each vector. The outputs are concatenated together to form a single output

sequence.

The self-attention layers are order-agnostic, and thus, the order of the frames in a clip is neglected. However, since the frames' order information is crucial, we have to ensure that our Transformer model considers the order of the frames within a clip . In order to achieve this, we use a positional embedding layer before the multi-head attention layer in which the position of each frame within a clip is encoded and added to the precomputed feature maps.

We adopt 1 single attention head in the training of our transformer encoder. The output of the multi-head attention layer is forwarded into a normalization layer. Then, the output is forwarded into a dense layer. Following that, the dense layer output is forwarded into a normalization layer. Finally, the results are pooled using a global max pooling layer, and forwarded into a fully connected layer for the CVS prediction.

3.4. Spatio-Temporal Region Graph Model

In this section, we describe, in detail, our proposed architecture for CVS prediction using spatio-temporal region graphs.

The aforementioned spatio-temporal models (e.g. LSTM) are able to capture the information encoded over time. Nonetheless, explicit fine-grained modeling of the surgical scene is a key limitation of these models. Moreover, the reasoning about spatial and semantic relationships between anatomical structures is fundamental for accurate CVS assessment, which is not explicit in our spatio-temporal models.

To address these limitations, we propose to utilize spatio-temporal region graph model which helps in modeling the surgical scene over time in a fine-grained manner. Our spatio-temporal surgical region graphs model is inspired by the work of Wang and Gupta (2018) where the authors developed a framework for video actions recognition.

The main components of the spatio-temporal region graphs architecture are as follows:

1. Inflated 3D CNN (I3D) for extracting spatio-temporal features.
2. Region Proposals Network (RPN) for extracting object proposals from each frame.
3. Region of Interest Pooling (ROI Pooling) for extracting ROI features from the I3D feature maps using the RPN proposals. These ROI features are nodes in the graph.
4. Two approaches to construct adjacency matrices relating these nodes, forward-backward graph and similarity graph.
5. Graph Convolutions Networks (GCNs) to process these graphs.

3.4.1. Region Proposal Network

The first step in building the graph is to extract candidate objects from each frame. In order to achieve this, Ren et al. (2015) proposed a Region Proposal Network (RPN) which helps to extract region proposals from a single image. We adopt the implementation of this method from Wu et al. (2019). Unlike the work of Wang and Gupta (2018), we have to retrain the RPN on our surgical dataset.

RPN Ground Truth. Our dataset "Endoscapes" contains only segmentation masks and CVS labels. However, to train an RPN network, we need to have bounding boxes ground truth. Therefore, we generate the bounding boxes ground truth based on the segmentation masks that we have using OpenCV and Numpy as follows:

1. Create a zero matrix (M) with size equal to the size of the segmentation mask.
2. Iterate through all labels except the background.
3. Copy the mask of the current label, referenced by the iteration loop, to M so that M contains only the segmentation mask of the current label (as ones) with zeros elsewhere.
4. Find the connected components in M so that we generate different bounding boxes for objects from the same class but are not connected spatially (e.g. the tool may have two instances in the same image).
5. Get the position of the pixels which contain the current label in the form (x1, y1, x2, y2), where x1, y1, x2, y2 denote for the upper-left, upper-right, lower-left, and lower-right position.
6. Save these bounding boxes in the dataset to represent our bounding boxes ground truth.

RPN training. Using the generated ground truth labels, we finetune a ResNet-50 pretrained RPN. The RPN outputs a number of region proposals from a single image ordered by the likelihood that the proposal is an object (objectness score). As a result, this allows us to limit the number of proposals used for our downstream tasks by taking the proposals whose objectness scores are above a certain threshold or by taking N proposals which have the highest objectness score.

3.4.2. Inflated 3D CNN

In order to extract feature maps from a clip, we use an Inflated 3D CNN (I3D) model Hara et al. (2018) which takes a clip as an input, and outputs feature maps of shape $B \times C \times T \times H \times W$, where B, C, T, H, W denote batch size, number of channels, time, image height, and image width, respectively.

Furthermore, we utilize an inflated 3D CNN which was previously pre-trained on ImageNet as a 2D CNN, and then inflated into 3D CNN and trained on Kinetics video dataset Kay et al. (2017). Our full I3D model architecture is illustrated in Figure 5.
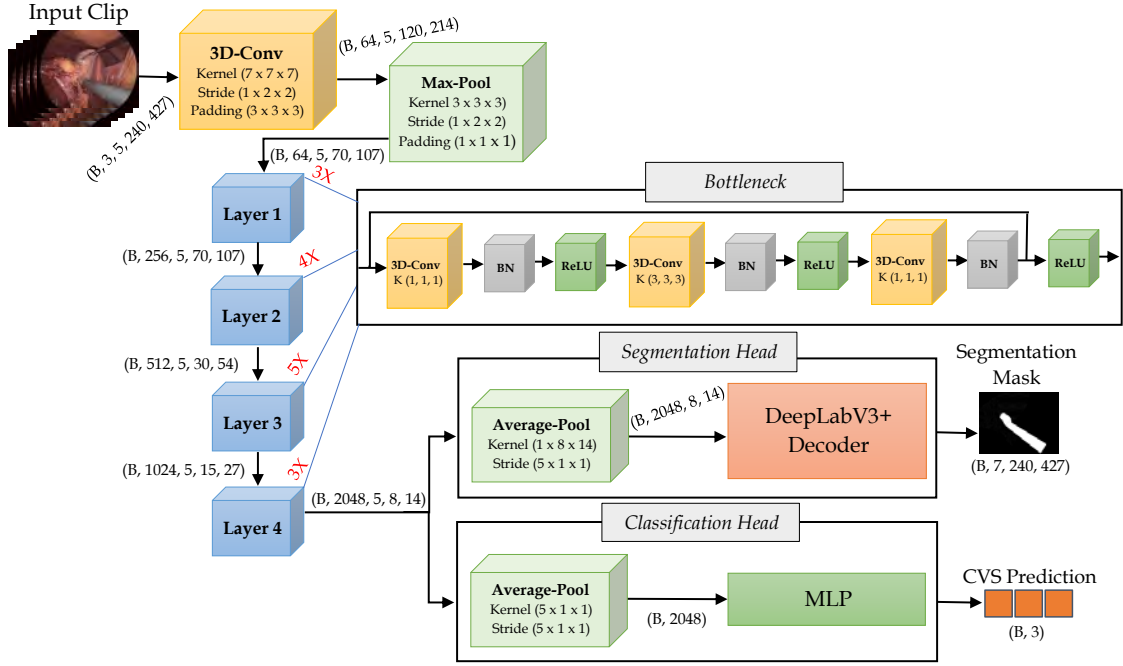
Figure 5: The architecture of the I3D model.

Clip Augmentation. We use the same image augmentations used in the training of the single-frame models. However, we apply the exact augmentation to all the frames within the same clip. We ensure this by randomizing the augmentation parameters before transforming the clip. Figure 6 visualizes the augmentation of 2 clips.

3.4.3. Region of Interest Pooling

Our I3D feature maps contain the features of the whole frames within a clip. However, we are only interested in some regions of these frames which are defined by the RPN proposals. The problem is that our RPN outputs objects bounding boxes in the resolution of the original images (e.g. 240×427), and our I3D model outputs feature maps that are down-sampled (e.g. 8×14). As a result, Ren et al. (2015) proposed a method called "Region of Interest Pooling (ROIPooling)" in which we can pool the ROI features from the I3D feature maps (low resolution) using the RPN bounding boxes (higher resolution) to a pre-defined size (e.g. 7×7). In this case, we obtain features of size $B \times T \times N \times C$ where B, T, N, C denote the batch size, time, number of proposals, and number of channels, respectively. To improve the precision of ROIPooling, He et al. (2017) proposed a similar approach called "ROIAlign" which avoids the quantization done by ROIPooling.

We adopt ROIAlign to extract features from each region of interest (ROI) which results in C×7×7 features from each ROI. After that, we forward these C×7×7

features into an average-pooling layer to output C×1×1 features.

3.4.4. Building the Graph

Having extracted object proposals and their features, we can now build our graph. Our graph is composed of nodes and edges where each node represents an object proposal, and each edge represents a connection/relationship between two nodes. In our case, the edge is represented by a value between [0,1] to describe if two nodes are connected in the graph.

To construct the edges in our graph, we follow two approaches, as introduced in Wang and Gupta (2018) as follows:

Forward-Backward Graph. This graph aims to link an object at time (**t**) with another object at time (**t + 1**) which guarantees that the connected objects are not only close to each other spatially, but also temporally. To build this graph, we utilize only the RPN object proposals by measuring the overlap between an object in a frame at time (**t**) with all other objects in a frame at time (**t + 1**). We can rely on the Intersection over Union (IoU) metric as formulated by Equation 1.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \qquad (1)$$

In this equation, **A** is the first bounding box, **B** is the second bounding box. However, if we do not have an overlap between an object **A** in a frame at time (**t**) with any other object in the following frame, the IoU
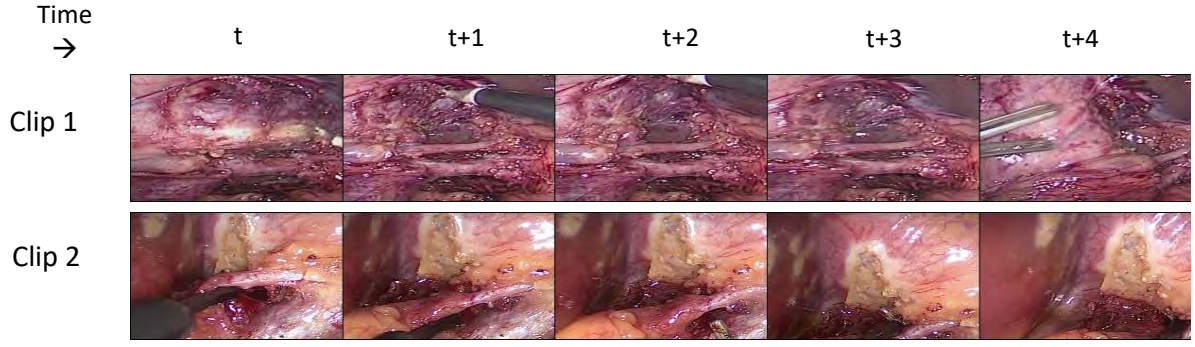
Figure 6: Illustration of similar augmentations being applied to all the frames within each clip.

value between the object **A** with the other objects will be 0 regardless of how far these objects from the object **A**. In this case, we will not be able to link this object with the closest one because there is no overlap. To address this, we propose to use a new variant of this metric namely "Generalized IoU (GIoU)" (Rezatofighi et al., 2019) which considers the distance between the two bounding boxes not only the overlap. This, however, allows us to build an edge between 2 objects even if they do not overlap. The GIoU is formulated in Equation 2.

$$GIoU = \frac{|A \cap B|}{|A \cup B|} - \frac{|C \backslash (A \cup B)|}{|C|} = IoU - \frac{|C \backslash (A \cup B)|}{|C|} \quad (2)$$

Here, **C** is the smallest convex hull that encloses both bounding boxes **A** and **B**.

We build this graph as follows:

1. Create a (M×M) zero matrix namely "adjacency matrix", where M is equal to the number of all object proposals within a clip. Each row corresponds to the edges values between an object with all other objects within a clip.

2. For each object proposal in a frame at time (**t**), calculate the GIoU score between that object with all other objects of the frame at time (**t + 1**).

3. Link the object at time (**t**) with the object at time (**t + 1**) whose GIoU score is greater than 0 by a direct edge in which we assign the GIoU score to the edge value (the intersection of the two objects in the adjacency matrix).

Additionally, we normalize each row by dividing each row by the sum of the row values. At the end, the obtained (M×M) matrix is used by our Graph Convolutional Network (GCN) as the adjacency matrix which is explained in Section 3.4.5. Furthermore, to enrich the graph representations, we also construct another graph similar to this one but the difference is that we move backward starting from the last frame within a clip. In other words, we compare the GIoU of an object at time

(**t**) with all other objects at time (**t − 1**)). We refer to the first graph as the forward graph, and this graph as the backward graph.

Similarity Graph. This graph aims to link an object in a frame at time (**t**) with the most similar object (based on a similarity measurement) in any frame within the same clip. In order to construct this graph, we utilize the pooled features (in the latent space) of each ROI to build an affinity matrix. Equation 3 describes our method of constructing this matrix, which is then used as the adjacency matrix of the GCN.

$$F\left(\mathbf{x}_i, \mathbf{x}_j\right) = \phi\left(\mathbf{x}_i\right)^T \phi'\left(\mathbf{x}_j\right) \quad (3)$$

Here, ϕ and ϕ' are two different transformations of the ROI features and can be obtained by $\phi(\mathbf{x}) = \mathbf{w}\mathbf{x}$ and $\phi'(\mathbf{x}) = \mathbf{w}'\mathbf{x}$. $\mathbf{x}$ denotes the ROI feature, whereas the weight parameters $\mathbf{w}$ and $\mathbf{w}'$ can be trained with backpropagation.

To implement this equation, we use 2 separate Multi Layer Perceptrons (MLPs) in which we forward the (B×M×C) features to both MLPs which results in 2 different tensors of the updated features with the same input shape (B×M×C), then we transpose the second tensor to obtain (B×C×M) tensor, and multiply both tensors to obtain a single (B×M×M) tensor. This resulting matrix is our affinity matrix containing the edge values of our proposals. However, the edge values at this point are not scaled to a consistent range. One way to do this is using a Softmax layer such that the sum of all edge values between a proposal at time (**t**) with all other proposals is equal to one.

This process results in an (M×M) adjacency matrix which connects two semantically related objects with each other regardless of their temporal location within a clip. In other words, an object in a frame at time (**t**) may be linked with another object in a frame at time (**t + 4**).

3.4.5. Graph Convolutional Networks

After we build the graphs, we obtain 3 adjacency matrices obtained from the forward graph, backward

graph, and the similarity graph in which the relationship between an object in a frame at time $(\mathbf{t} + \mathbf{4})$ with all other objects within a clip is quantified by the edge values. By looking at one of these matrices, one can realize that there is a relationship between an object with another object whose edge value is the highest. In this case, we can recognize similar objects even though they are far away from one another spatially and temporally. This brings us to the question: Can we train a model that takes into consideration the adjacency matrix to consider only the related objects and ignore unrelated objects? Fortunately, the answer is yes, we can utilize Graph Convolutional Networks (GCN) which operates on graphs. To design our GCN, we adopts the model proposed by Kipf and Welling (2016) which takes as an input the ROI features of shape (B×M×C) and outputs the refined features with the same input shape (B×M×C).

3.4.6. Spatio-Temporal Region Graph Architecture

After building and preparing all the aforementioned components of our architecture, we connect these components as illustrated in illustrated in Figure 7. Besides the GCNs features, we also pool the I3D features and concatenate both features and forward them into a fully connected (FC) layer for CVS prediction. Additionally, we also incorporate a segmentation head in which we utilize the DeepLabV3+ decoder for segmentation masks prediction. As a result, our model can be trained in a multi-task manner for the objective of both CVS and segmentation as well as in a single-task manner for the objective of CVS only by detaching the segmentation head.

3.5. Loss Function

We use the binary cross entropy (BCE) loss as our loss function which measures how far or close the predicted logits from the true labels. BCE is formulated in Equation 4.

$$L_{BCE} = -\frac{1}{n} \sum_{i=1}^{n} \left(Y_i \cdot \log \hat{Y}_i + (1 - Y_i) \cdot \log \left(1 - \hat{Y}_i \right) \right)$$
(4)

Here, n denotes the number of samples, Y the true labels, and $\hat{Y}$ the predicted labels.

To mitigate the class imbalance, we calculate the class weights we used inverse frequency balancing to compute class weights (Ahmed et al., 2020).

Label	C1	C2	C3
Positive	3.15	4.32	2.57
Negative	0.59	0.56	0.62

Table 2: The weights of the positive and negative classes of each label (C1, C2, and C3).

This process results in 2 vectors each with length equal to the number of classes 3×1 which contain the weights of the positive/negative labels for each class as shown in Table 2. After that, the vector containing the positive weights is forwarded to the BCE loss function to consider the class imbalance by penalizing the positive predictions with respect to the precomputed weights.

Multitask Loss Weights. We tune the weights of both classification and segmentation loss functions which results in using 0.25 for the classification loss and 1 for the segmentation loss. Therefore, the final loss function is formulated in Equation 5.

$$L_{final} = 0.25 \times L_{\text{class}} + L_{\text{seg}}$$
(5)

RPN Loss Functions. We use two loss functions to train our RPN network as follows:

1. Objectness loss: we adopt BCE loss to minimize the error of the objectness prediction of the proposals.
2. Localization loss: to localize objects, we use L1 loss function as formulated by Equation. 6 which helps to minimize the error by computing the sum of all the absolute differences between the true label and the predicted label. This loss is computed only when the ground truth objectness score is 1 (foreground).

$$\text{L1} = \sum_{i=1}^{n} \left| y_{\text{true}} - y_{\text{predicted}} \right|$$
(6)

3.6. Training setup

The training of the models is performed on 1 single NVIDIA 24 GB RTX 6000 GPU. We train each model for 100 epochs. Adam optimizer is selected with a learning rate of 1e-4. The batch size is 8. The input frames are resized to 240×427. We adopt the same data augmentation performed by the original work (DeepCVS baseline) which include (Random Crop, Random Resize, Random Brightness with brightness factor = 0.2). The image channels are normalized to zero mean and standard deviation = 1 by subtracting and dividing each input channel by (0.5).

3.7. Implementation Details

I3D training. We train our I3D backbone on the Endoscapes dataset (the Temporal dataset) using the following strategies:

- Single-task CVS prediction using the 11090 CVS labels (I3D-CVS).

- Single-task segmentation masks prediction using the 1933 segmentation masks (I3D-Seg).
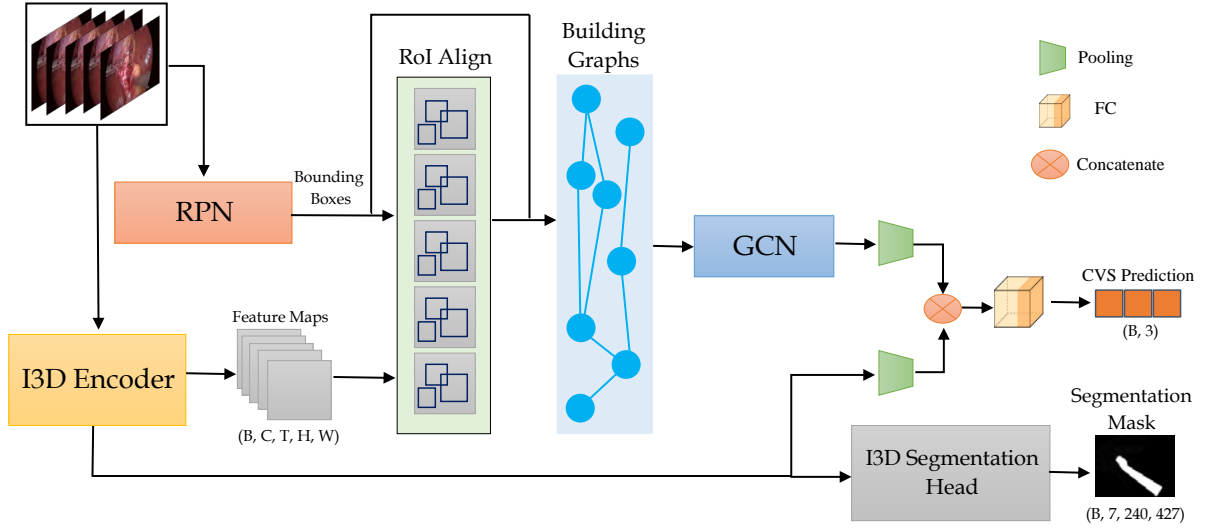
Figure 7: The architecture of our proposed Spatio-Temporal Region Graph Model.

Table 3: Results of the CVS prediction (mAP (%) for CVS, Macro F1 (%) for Segmentation). ST-DeepCVS refers to the DeepCVS with a GRU layer.

Method		Training Objective	Test Performance	
			CVS	**Seg**
Single Frame baseline (DeepCVS)		CVS	39.60	-
		CVS + Seg	54.07	68.57
Spatio-Temporal (**ours**)	ST-DeepCVS	CVS + Seg	**60.92**	68.57
	I3D	CVS	38.71	-
		Seg	-	65.81
		CVS + Seg	57.37	65.24
	Region Graphs	CVS + Seg	59.55	**68.80**

- Multi-task CVS and segmentation masks predictions using the 1933 segmentation masks and their counterpart 1933 CVS labels (I3D-MT).

- Multi-task CVS and segmentation masks predictions using the 11090 segmentation masks and their counterpart 11090 CVS labels. Since we have only 1933 segmentation masks, we generate pseudo labels using our trained DeepLabV3+ for the missing 9157 segmentation masks (I3D-Pseudo-MT).

The highest performing model is chosen to be used in our proposed method. When our training objective is the CVS prediction, we ignore the segmentation head. Similarly, we ignore the classification head when our training objective is predicting segmentation masks. We only use both heads (segmentation and classification) with multi-task training.

Spatio-Temporal Region Graph Model training. In the beginning, we freeze the RPN and the I3D model

to train the GCN, the Similarity Graph module (considering that our Similarity graph is learnt by training unlike the forward and backward graphs), and the fully connected layer for 10 epochs with 1e-3 learning rate. Following that, the I3D model is unfrozen and trained with the full network end-to-end but we still keep the RPN weights frozen. At this stage, we train the model with a learning rate = 1e-5.

3.8. Ablation study setup

Sequence Length. In all of our experiments with spatio-temporal models, we adopt 5 frames at 1fps for each clip. Now, we conduct experiments while increasing the number of frames within a clip to (10, 15, 50, 100, 200, 500). This means that we will consider the temporal dependencies of more frame to predict CVS labels. We use zero-padding for the clips at the beginning of each video to satisfy the clip length. Furthermore, we consider also increasing the frame rate from 1fps to 5fps and 25fps.

Table 4: Results of the I3D model (mAP for CVS, Macro F1 for Segmentation)

Training Strategy	Training Samples	Test Performance	
		CVS	Seg
I3D-CVS	11090	38.71	-
I3D-Seg	1933	-	**65.81**
I3D-MT	1933	41.61	64.36
I3D-Pseudo-MT	11090	**57.37**	65.24

Table 5: Segmentation Results of the I3D model with different frame rate per second

Frame Rate	1 fps	5 fps	25 fps
Macro F1	**65.81**	62.42	62.54

4. Results

4.1. Quantitative Results

We evaluated the performance of each model on the test set of the Endoscapes data set using the metrics explained in Section 3.9. Table 3 shows the results of each model for both CVS prediction and hepatocystic anatomy and tools segmentation. On the other hand, Figure 8 compares the performance of each temporal layer within the ST-DeepCVS model against the clip length.

Table 4 shows the results of the I3D model using different training strategies with different training objectives. Table 5 shows the results of the I3D model trained in a single-task manner for the objective of segmentation only when using more frame rate per second.

Furthermore, the detailed performance of our proposed methods on each CVS criterion based on the average precision and balanced accuracy is shown in Table 6.

4.2. Qualitative Results

The proposed models are also evaluated qualitatively by visualizing the models' predictions. Figure 9 illustrates the CVS prediction labels of 3 random samples from the the Endoscapes' test set and compares these predictions with the ground truth.

Furthermore, we also evaluated the performance of some components of our ST-Graph model such at the region proposal network (RPN), the construction of the forward and backward graphs. Figure 10 illustrates the object proposals extracted by the RPN from 2 different test images, whereas Figure 11 shows both the forward and backward graphs built using the RPN object proposals for 1 clip composed of 5 frames.

5. Discussion

5.1. Training Objective

The Endoscapes dataset contains CVS labels and segmentation masks of the hepatocystic anatomy with surgical tools. The assessment of the critical view of safety in laparoscopic cholecystectomy can be performed by identifying anatomical landmarks from the surgical view. Table 3 demonstrates the importance of identifying hepatocystic landmarks (using hepatocystic segmentation masks) to achieve more accurate CVS prediction as we obtain mAP of 39.60% when training DeepCVS for the objective of CVS only whereas we obtain 54.07% when training for both CVS and segmentation objectives (14.47% boost). Moreover, training with

Frame Rate In our experiments, we adopt 1 frame per second. However, we also examine the frame rate choice by performing experiments on the I3D model with different frame rates (1fps, 5fps, and 25fps).

3.9. Metrics and Assessment

The models' performance is evaluated following the evaluation done by the state-of-the-art for each task.

CVS criteria prediction. We rely on the mean average precision (mAP) and the balanced accuracy metric to compare our algorithms against state-of-the-art. The mean average precision has shown to be optimal for multilabel classification tasks, and is formulated by Equation. 7.

$$\text{mAP} = \sum_n (R_n - R_{n-1}) P_n \qquad (7)$$

In this formula, R is the recall (Equation. 8), and P is the precision (Equation. 9), and n is the number of samples.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \qquad (8)$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \qquad (9)$$

Since our dataset is characterized with significant class imbalance, we also rely on the balanced accuracy metric. It is calculated by computing the average of recall obtained on each class.

Segmentation . We test the performance of our segmentation results based on macro F1 score that balances the precision and recall on the positive class as shown in Equation 10. The IoU score is also adopted by the state-of-the-art to assess the segmentation results. However, since our main objective is the CVS assessment, we ignore this metric and rely only on the F1 score.

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (10)$$

Figure 8: The performance of the temporal DeepCVS models with different clip length.

Table 6: Detailed CVS performance of St-DeepCVS and ST-Graph.

Criterion	AP (%)		Balanced Accuracy (%)	
	ST-DeepCVS	ST-Graph	ST-DeepCVS	ST-Graph
C1: Two Structures	62.56	61.87	75.89	73.22
C2: Hepatocystic Triangle	54.76	55.25	70.79	69.31
C3: Cystic Plate	65.44	61.55	71.19	68.05
Overall	60.92	59.55	72.62	70.19

the segmentation also helps the I3D model (18.66% boost).

5.2. ST-DeepCVS

In this study, spatio-temporal deep learning models to evaluate the criteria defining the critical view of safety are developed. As shown in Table 3, utilizing the temporal information encoded over time is helpful as it can be seen that the mAP improves from 54.07% when using the single frame DeepCVS model to 60.92% by just incorporating a single recurrent layer (6.85% boost).

Additionally, Figure 9 demonstrates that our ST-DeepCVS model was able to correctly predict all of the CVS criteria for 3 random samples from the dataset.

5.3. Clip Length

One of the most significant hyper-parameters in sequence models is the sequence length. In our study, we adopt 5 frames per clip in the training of most of our spatio-temporal models. However, we also examined the performance of increasing the clip length as illustrated in Figure 8. We can see that TCN was the most robust model to clip length increase. On the other

hand, we realize a significant drop in performance in the Transformers model. We attribute this drop in performance to the fact that Transformers require significant tuning comparing to recurrent layers (e.g. LSTM). Moreover, we can see that most temporal models were close to each other (by ≈ 1 - 4% mAP) when using (5-15) frames unlike RNNs which appear to perform poorly in the CVS prediction even with short clip length.

5.4. Inflated 3D CNN

5.4.1. Training with pseudo-labels

Table 4 demonstrates that by increasing the training set from 1933 to 11090 by utilizing the segmentation pseudo-labels in a semi-supervised manner when training the I3D model, we are able to improve the performance of the CVS prediction from 38.71% to 57.37% (18.66% boost). This highlights the effectiveness of semi-supervised learning when the training data as limited.

5.5. Frame Rate

To ensure the proper frame rate, we examined the performance of the I3D model with different frame

Model	C1	C2	C3	C1	C2	C3	C1	C2	C3
ST-DeepCVS (ours)	1	1	1	0	1	1	1	0	0
ST-Graph (ours)	1	1	1	0	1	1	1	0	0
I3D-Pseudo-MT (ours)	1	0	1	0	0	1	1	0	0
I3D-CVS (ours)	0	0	1	1	0	0	0	0	1
DeepCVS (CVS +Seg)	1	1	0	1	0	1	1	0	1
DeepCVS (CVS)	1	0	0	0	0	1	0	0	0
Ground Truth	1	1	1	0	1	1	1	0	0

Figure 9: Comparison between our proposed models with the baselines in CVS prediction for 3 different input endoscopic images (To be replaced with the actual values later).
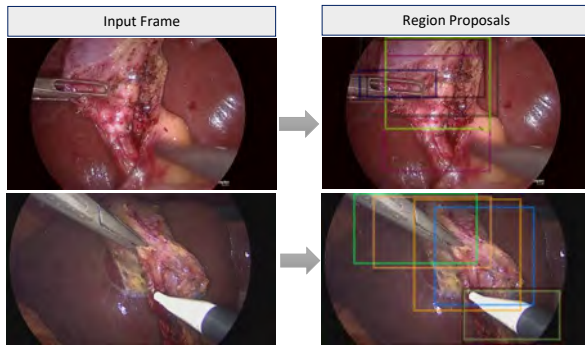
Figure 10: Qualitative results of the RPN.

rate. Table 5 demonstrates that the spatio-temporal model (I3D) performs better with 1 fps than 5 fps or 25 fps which indicates that more relevant information are learnt by the model with 1 fps. Therefore, we maintain this frame rate in our future experiments.

5.5.1. RPN

The qualitative results of our RPN shown in Figure 10 demonstrate that we are able to obtain relevant proposals of the region of interests which can help building robust graph. We also realize from the top-right image that one of the tools was not detected by the RPN which can be simply solved by increasing the number of proposals as we only visualized 6 proposals but during the training of our graph-based model we extract 16 proposals.

5.6. Building the Graph

The visualization of the forward and backward graph in Figure. 11 shows that we can link the same objects in different frames. However, we realize that this approach is not efficient when we have overlapped bounding boxes that are bigger than the object's bounding box. As a result, we should investigate a proper approach to tackle this limitation.

5.7. ST-Graph

As shown in Table 3, our ST-Graph model outperformed the DeepCVS baseline in both CVS prediction (5.48% boost) and segmentation (0.23% boost). Comparing with the I3D-Seg model, we realize that building the graph has improved the segmentation performance by (2.99%).

utilizing the temporal information encoded over time is helpful as it can be seen that the mAP improves from 54.07% when using the single frame DeepCVS model to 60.92% by just incorporating a single recurrent layer (6.85% boost).

Furthermore, we can see from Figure 9 that our ST-Graph model was able to correctly predict all of the CVS criteria for 3 random samples from the dataset.

5.8. CVS Criteria

We realize from the detailed performance of our proposed methods, shown in Table 6, that the Hepatocystic Triangle criterion was more challenging (scoring less mAP) for our methods to predict.

6. Conclusions

In this study, two deep learning approaches to evaluate the critical view of safety were developed, specifically focusing on spatio-temporal methods. Our first approach incorporates temporal layers on top of the Deep-CVS model to extract spatio-temporal features from
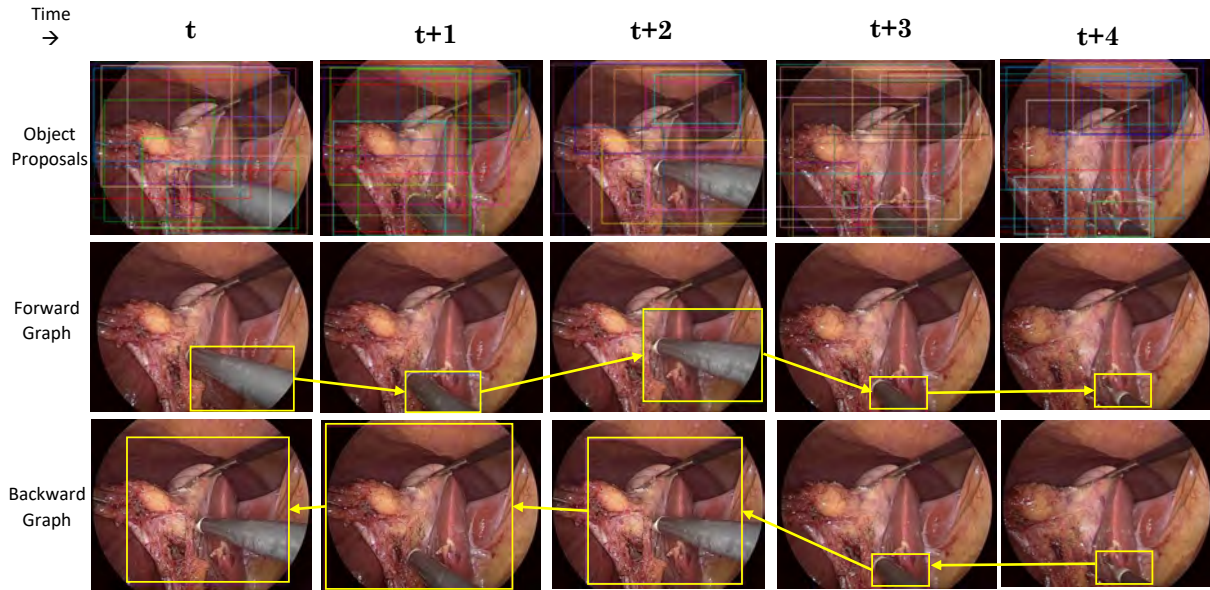
Figure 11: Qualitative results of the Forward and Backward Graphs. We show the edge of only 1 object for simplicity. Other objects are linked by direct edges in the same way.

Endoscopic clips. The second approach builds spatio-temporal region graphs to model the LC surgical scene which, in turn, helps in understanding the surgical scene at a fine-grained level and has improved the segmentation performance. Both proposed methods outperformed the baseline in CVS prediction. Further extension of this work may involve building different graphs to address the limitations of the graph building approaches and improve the performance.

7. Future Work

In this section, we present the future experiments that will be conducted to further validate our approaches.

I3D Backbone. After selecting the best training strategy for our I3D model, it is interesting to also experiment architectures such as MobileNet or InceptionV3 other than the ResNet-50. We leave this experiment for future work due to the limited time we have.

Spatio-Temporal Graph Model. Since we utilize 3 types of graphs, it is interesting to check how much each graph helps in CVS prediction by training the model each time with one graph. Furthermore, it is also interesting to examine the performance of our ST-Graph method when trained for a single task with the objective of CVS prediction.

Acknowledgments

References

Ahmed, N., Dilmaç, F., Alpkocak, A., 2020. Classification of biomedical texts for cardiovascular diseases with deep neural network using a weighted feature representation method, in: Healthcare, Multidisciplinary Digital Publishing Institute. p. 392.

Alapatt, D., Mascagni, P., Vardazaryan, A., Garcia, A., Okamoto, N., Mutter, D., Marescaux, J., Costamagna, G., Dallemagne, B., Padoy, N., 2021. Temporally constrained neural networks (tcnn): A framework for semi-supervised video semantic segmentation. arXiv preprint arXiv:2112.13815 .

Alshirbaji, T.A., Jalal, N.A., Docherty, P.D., Neumuth, T., Möller, K., 2021. A deep learning spatial-temporal framework for detecting surgical tools in laparoscopic videos. Biomedical Signal Processing and Control 68, 102801.

Bai, S., Kolter, J.Z., Koltun, V., 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 .

Berci, G., Hunter, J., Morgenstern, L., Arregui, M., Brunt, M., Carroll, B., Edye, M., Fermelia, D., Ferzli, G., Greene, F., et al., 2013. Laparoscopic cholecystectomy: first, do no harm; second, take care of bile duct stones.

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), pp. 801–818.

Chen, V.S., Varma, P., Krishna, R., Bernstein, M., Re, C., Fei-Fei, L., 2019. Scene graph prediction with limited labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2580–2590.

Cheng, K., You, J., Wu, S., Chen, Z., Zhou, Z., Guan, J., Peng, B., Wang, X., 2022. Artificial intelligence-based automated laparoscopic cholecystectomy surgical phase recognition and analysis. Surgical endoscopy 36, 3160–3168.

Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 .

Conrad, C., Wakabayashi, G., Asbun, H.J., Dallemagne, B., Demartines, N., Diana, M., Fuks, D., Giménez, M.E., Goumard, C., Kaneko, H., et al., 2017. Ircad recommendation on safe laparoscopic cholecystectomy. Journal of Hepato-Biliary-Pancreatic Sciences 24, 603–615.

da Costa Rocha, C., Padoy, N., Rosa, B., 2019. Self-supervised surgical tool segmentation using kinematic information, in: 2019 International Conference on Robotics and Automation (ICRA), IEEE. pp. 8720–8726.

Czempiel, T., Paschali, M., Keicher, M., Simson, W., Feussner, H., Kim, S.T., Navab, N., 2020. Tecno: Surgical phase recognition with multi-stage temporal convolutional networks, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 343–352.

Czempiel, T., Paschali, M., Ostler, D., Kim, S.T., Busam, B., Navab, N., 2021. Opera: Attention-regularized transformers for surgical phase recognition, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 604–614.

Gao, X., Jin, Y., Long, Y., Dou, Q., Heng, P.A., 2021. Trans-svnet: accurate phase recognition from surgical videos via hybrid embedding aggregation transformer, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 593–603.

Guédon, A.C., Meij, S.E., Osman, K.N., Kloosterman, H.A., van Stralen, K.J., Grimbergen, M., Eijsbouts, Q.A., van den Dobbelsteen, J.J., Twinanda, A.P., 2021. Deep learning for surgical phase recognition using endoscopic videos. Surgical Endoscopy 35, 6150–6157.

Halle-Smith, J.M., Hodson, J., Stevens, L.G., Dasari, B., Marudanayagam, R., Perera, T., Sutcliffe, R.P., Muiesan, P., Isaac, J., Mirza, D.F., et al., 2019. A comprehensive evaluation of the long-term economic impact of major bile duct injury. HPB 21, 1312–1321.

Hara, K., Kataoka, H., Satoh, Y., 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 6546–6555.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn, in: Proceedings of the IEEE international conference on computer vision, pp. 2961–2969.

Islam, M., Atputharuban, D.A., Ramesh, R., Ren, H., 2019. Real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning. IEEE Robotics and Automation Letters 4, 2188–2195.

Johnson, J., Gupta, A., Fei-Fei, L., 2018. Image generation from scene graphs, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1219–1228.

Kadkhodamohammadi, A., Luengo, I., Stoyanov, D., 2022. Patg: position-aware temporal graph networks for surgical phase recognition on laparoscopic videos. International Journal of Computer Assisted Radiology and Surgery 17, 849–856.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al., 2017. The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 .

Khan, S., Cuzzolin, F., 2021. Spatiotemporal deformable scene graphs for complex activity detection .

Khan, S., Naseer, M., Hayat, M., Zamir, S.W., Khan, F.S., Shah, M., 2021. Transformers in vision: A survey. ACM Computing Surveys (CSUR) .

Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 .

Kondo, S., 2021. Lapformer: surgical tool detection in laparoscopic surgical video using transformer architecture. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization 9, 302–307.

Li, L., Li, X., Ding, S., Fang, Z., Xu, M., Ren, H., Yang, S., 2022. Sirnet: Fine-grained surgical interaction recognition. IEEE Robotics and Automation Letters 7, 4212–4219.

Liu, K., Zhao, Z., Shi, P., Li, F., Song, H., 2022. Real-time surgical tool detection in computer-aided surgery based on enhanced feature fusion convolutional neural network. Journal of Computational Design and Engineering .

Madani, A., Namazi, B., Altieri, M.S., Hashimoto, D.A., Rivera, A.M., Pucher, P.H., Navarrete-Welton, A., Sankaranarayanan, G., Brunt, L.M., Okrainec, A., et al., 2022. Artificial intelligence for intraoperative guidance: using semantic segmentation to identify surgical anatomy during laparoscopic cholecystectomy. Annals of surgery .

Mascagni, P., Fiorillo, C., Urade, T., Emre, T., Yu, T., Wakabayashi, T., Felli, E., Perretta, S., Swanstrom, L., Mutter, D., et al., 2020. Formalizing video documentation of the critical view of safety in laparoscopic cholecystectomy: a step towards artificial intelligence assistance to improve surgical safety. Surgical endoscopy 34, 2709–2714.

Mascagni, P., Vardazaryan, A., Alapatt, D., Urade, T., Emre, T., Fiorillo, C., Pessaux, P., Mutter, D., Marescaux, J., Costamagna, G., et al., 2022. Artificial intelligence for surgical safety: automatic assessment of the critical view of safety in laparoscopic cholecystectomy using deep learning. Annals of surgery 275, 955–961.

Mittal, G., Agrawal, S., Agarwal, A., Mehta, S., Marwah, T., 2019. Interactive image generation using scene graphs. arXiv preprint arXiv:1905.03743 .

Monasterio-Exposito, L., Pizarro, D., Macias-Guarasa, J., 2022. Label augmentation to improve generalization of deep learning semantic segmentation of laparoscopic images. IEEE Access 10, 37345–37359.

Namazi, B., Sankaranarayanan, G., Devarajan, V., 2022. A contextual detector of surgical tools in laparoscopic videos using deep learning. Surgical endoscopy 36, 679–688.

Ni, Z.L., Bian, G.B., Li, Z., Zhou, X.H., Li, R.Q., Hou, Z.G., 2022a. Space squeeze reasoning and low-rank bilinear feature fusion for surgical image segmentation. IEEE Journal of Biomedical and Health Informatics .

Ni, Z.L., Zhou, X.H., Wang, G.A., Yue, W.Q., Li, Z., Bian, G.B., Hou, Z.G., 2022b. Surginet: Pyramid attention aggregation and class-wise self-distillation for surgical instrument segmentation. Medical Image Analysis 76, 102310.

Nwoye, C.I., Alapatt, D., Yu, T., Vardazaryan, A., Xia, F., Zhao, Z., Xia, T., Jia, F., Yang, Y., Wang, H., et al., 2022a. Cholectriplet2021: A benchmark challenge for surgical action triplet recognition. arXiv preprint arXiv:2204.04746 .

Nwoye, C.I., Gonzalez, C., Yu, T., Mascagni, P., Mutter, D., Marescaux, J., Padoy, N., 2020. Recognition of instrument-tissue interactions in endoscopic videos via action triplets, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 364–374.

Nwoye, C.I., Yu, T., Gonzalez, C., Seeliger, B., Mascagni, P., Mutter, D., Marescaux, J., Padoy, N., 2022b. Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos. Medical Image Analysis 78, 102433.

Owen, D., Grammatikopoulou, M., Luengo, I., Stoyanov, D., 2021. Detection of critical structures in laparoscopic cholecystectomy using label relaxation and self-supervision, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 321–330.

Pucher, P.H., Brunt, L.M., Fanelli, R.D., Asbun, H.J., Aggarwal, R., 2015. Sages expert delphi consensus: critical factors for safe surgical practice in laparoscopic cholecystectomy. Surgical endoscopy 29, 3074–3085.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28.

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S., 2019. Generalized intersection over union: A metric and a loss for bounding box regression, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 658–666.

Schreuder, A.M., Busch, O.R., Besselink, M.G., Ignatavicius, P., Gulbinas, A., Barauskas, G., Gouma, D.J., van Gulik, T.M., 2020. Long-term impact of iatrogenic bile duct injury. Digestive surgery 37, 10–21.

Seenivasan, L., Islam, M., Ng, C.F., Lim, C.M., Ren, H., 2022. Biomimetic incremental domain generalization with a graph network for surgical scene understanding. Biomimetics 7, 68.

Sestini, L., Rosa, B., De Momi, E., Ferrigno, G., Padoy, N., 2022. Fun-sis: a fully unsupervised approach for surgical instrument segmentation. arXiv preprint arXiv:2202.08141 .

Sharghi, A., Haugerud, H., Oh, D., Mohareri, O., 2020. Automatic operating room surgical activity recognition for robot-assisted surgery, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 385–395.

Shimizu, T., Hachiuma, R., Kajita, H., Takatsume, Y., Saito, H., 2021. Hand motion-aware surgical tool localization and classification from an egocentric camera. Journal of Imaging 7, 15.

Strasberg, S.M., 1995. An analysis of the problem of biliary injury during laparoscopic cholecystectomy. J Am Coll Surg 180, 101–125.

Strasberg, S.M., Brunt, M.L., 2010. Rationale and use of the critical view of safety in laparoscopic cholecystectomy. Journal of the American College of Surgeons 211, 132–138.

Tokuyasu, T., Iwashita, Y., Matsunobu, Y., Kamiyama, T., Ishikake, M., Sakaguchi, S., Ebe, K., Tada, K., Endo, Y., Etoh, T., et al., 2021. Development of an artificial intelligence system using deep learning to indicate anatomical landmarks during laparoscopic cholecystectomy. Surgical endoscopy 35, 1651–1658.

Törnqvist, B., Strömberg, C., Persson, G., Nilsson, M., 2012. Effect of intended intraoperative cholangiography and early detection of bile duct injury on survival after cholecystectomy: population based cohort study. Bmj 345.

Tunstall, L., von Werra, L., Wolf, T., 2022. Natural Language Processing with Transformers. " O'Reilly Media, Inc.".

Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M., Padoy, N., 2016. Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE transactions on medical imaging 36, 86–97.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

Wang, X., Gupta, A., 2018. Videos as space-time region graphs, in: Proceedings of the European conference on computer vision (ECCV), pp. 399–417.

Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R., 2019. Detectron2. https://github.com/facebookresearch/detectron2.

Xue, Y., Liu, S., Li, Y., Wang, P., Qian, X., 2022. A new weakly supervised strategy for surgical tool detection. Knowledge-Based Systems 239, 107860.

Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D., 2018. Graph r-cnn for scene graph generation, in: Proceedings of the European conference on computer vision (ECCV), pp. 670–685.

Yang, Z., Simon, R., Linte, C., 2022. A weakly supervised learning approach for surgical instrument segmentation from laparoscopic video sequences, in: Medical Imaging 2022: Image-Guided Procedures, Robotic Interventions, and Modeling, SPIE. pp. 412–417.

Yuan, K., Holden, M., Gao, S., Lee, W.S., 2021. Surgical workflow anticipation using instrument interaction, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 615–625.

Zhang, Z., Rosa, B., Nageotte, F., 2021. Surgical tool segmentation using generative adversarial networks with unpaired training data.

IEEE Robotics and Automation Letters 6, 6266–6273.

Zhao, Z., Jin, Y., Heng, P.A., 2022. Trasetr: Track-to-segment transformer with contrastive query for instance-level instrument segmentation in robotic surgery. arXiv preprint arXiv:2202.08453 .

Color consistency in clinical skin images

Manuel Ojeda Osorio, Bart Diricx and Johan Rostang

Kortrijk, Belgium

Abstract

Color consistency in clinical skin images is important because it allows dermatologists to monitor the evolution of skin conditions over time, to follow-up a medical treatment, to evaluate if the patient's skin needs treatment, among other reasons. Color variability may be originated from both the acquisition and visualization device which can be solved by calibrating both devices. Even though both devices are calibrated, there can be color variability because the devices' technical characteristics, different environment conditions (sunlight or variability of the scene illumination), etc. ICC color profiles are useful to perform an end-to-end calibration, this paper presents a pipeline which integrates a method for calibrating the color of images into an ICC workflow to maintain color consistency between different devices. The evaluation to determine how the ICC color profiles perform is done based on the equation CIE DeltaE2000 metric.

Keywords: Color management, Color consistency, Color profile, Medical photography, Medical imaging

1. Introduction

Color is a phenomenon of light, an interaction between light, object and viewer; it is one way how people differentiate identical objects. Color in images can evoke different emotions which is why many companies in different industries dedicate resources to research and develop new techniques to take advantage of colors, such as marketing (X-Rite, 2004).

Light is an electromagnetic wave, a form of radiation. There are different types of electromagnetic waves with different properties, such as X rays, microwaves or radio waves (Sliney, 2016). Different organisms can see different types of waves, spiders can see ultraviolet and reptiles infra-red light, the type of electromagnetic wave humans can see is called visible light (CJ Kazilek, 2009).

The visible light is an electromagnetic wave, the range of wavelengths within this visible spectrum is from about 400 to about 750 nm (Sliney, 2016). Humans only see this range of wavelengths because of the cells in our eyes, cone-shaped cells, which act as receivers for only that band of the spectrum. There are three types of cones with different sensitivity to light of different wavelengths: short (S), medium (M) and long (L), referred as "blue", "green", "red" (Dale Purves, 2001).
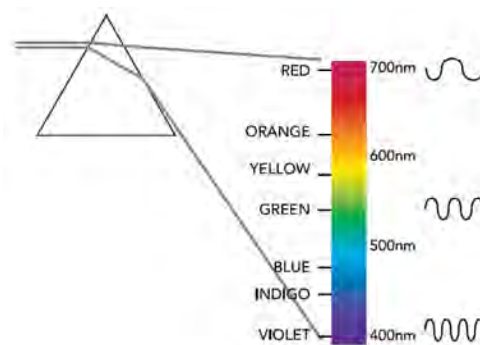
Figure 1: Representation of a beam of white light through a prism (X-Rite, 2004).

The combination of different wavelengths of light gives as result different colors. One example is passing a beam of white light through a prism which disperses the light, humans see different colors because our eyes respond to each individual wavelength (Leelakrishnan, 2022). Another example is one of the demonstrations James Clerk Maxwell did where he used red, green and blue filters and black and white pictures of colorful objects. When he projected the pictures through the filters, the original colorful objects could be seen; not only red,

green and blue showed up, but the oranges and yellows and purples as well. The last example also explains how color photography works, both film-based and digital (Sack, 2016).

As told, the object has an important role on the colors humans see along with a specific feature of the light, reflection. When light hits a surface of an object, depending on the material, some of the light is absorbed and the rest is reflected. All light that hits the eye is reflected light and the wavelength of the reflected light determines the color humans perceive (X-Rite, 2004).

The purpose of photography is to capture the images our eyes perceive, either physically with paper or digitally with computer files. This is achieved by mimicking the functioning of the eye with a device well known by everyone, the camera. Through the years, like everything, the camera has evolved from big and fixed devices to very small and portable devices which can be components of other devices, such as smartphones. Not only the physical characteristics have changed, but also the functional ones. The cameras used to capture a fixed scene in minutes, now are capable of capturing a sequence of scenes which are called videos; or scenes which for the human eye may be impossible to visualize, such achieved by high-speed cameras (Rick, 2013).
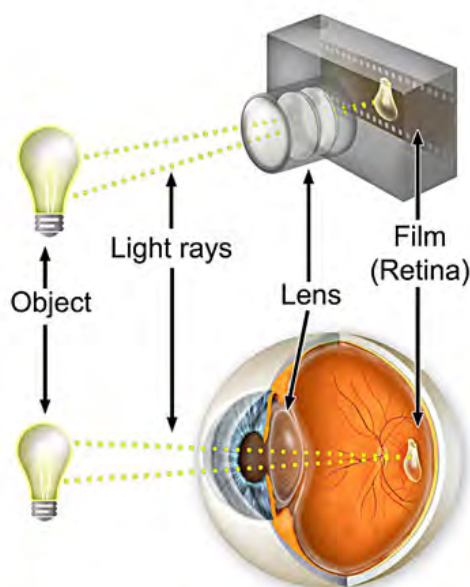
Figure 2: Basic diagram of how a camera mimics the human eye (Plaza, 2013).

Just as human eyes, cameras are different from each other. They might be the same model, but each camera captures the light in a different way. Acquisition variability will be the combination of the variability of the acquisition device and the variability of the scene, mainly caused by variability of the scene illumination. Regardless of the acquisition variability, one main detail of cameras is not being able, yet, to capture the same

range of colors as humans see, different cameras can capture different range of colors (Dunlop, 2022).

Not only the characteristics of the camera will define how the scene will be seen, but the method we use to visualize it, for example an image printed on paper or displayed on a screen. Just as cameras, devices or methods used to display an image will affect how humans perceive the image. The screens have very important contributors such as the range of colors they can display, contrast, ambient light, maximum luminance, because there is no device which can display the same range of colors that humans can see.

Technology advances have made possible to have different devices for different purposes, in order to visualize the information we need. Nowadays they are coming in different sizes for different devices, such as tablets, smartphones, laptops, smart watches, etc. The variety of devices generates the visualization variability mentioned before. R. Sharan and Iyer describe the displays as not being able to stand on their own, but a part of an information system and give a brief explanation of how the display has evolved from the Cathode-Ray-Tube (CRT) invented by Braun in 1897 to some of today's technologies depending on the device, like TV, cell phone or computer.

Figure 3: Example of how many displays can be used at once (Paul, 2022).

Gandhi (2015) explains briefly various display technologies. LCD displays generate an image from an internal light source through a liquid-crystal material to either block or transmit light. Plasma displays work by filling the region between two glass plates with a combination of gases; a series of firing voltages cause the gas to break down into a glowing plasma of electrons and ions. LED displays are a matrix of diodes arranged to form the pixel positions in the display. Flexible displays are flexible OLED, based on flexible substrate which can be either plastic, metal or flexible glass. Gandhi (2015) also suggests using the technology based on the necessity because each one of them has its own advantages and disadvantages.

Due to those differences of technologies, capabilities and people's needs, companies like Barco (2022) offer

a wide variety of display technologies for different sectors based on the needs such as entertainment with projectors or health sector with medical displays, etc.
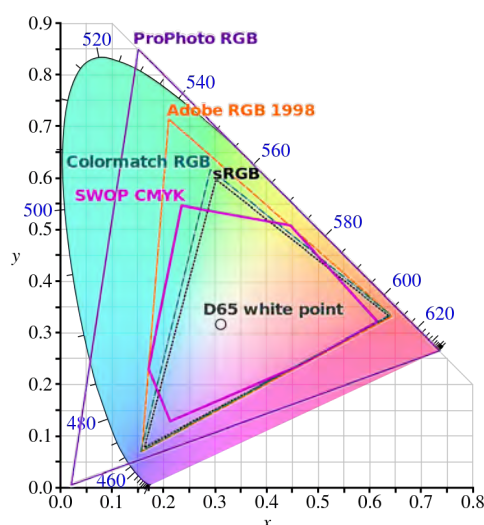
Figure 4: Comparison between color spaces and the visible spectrum.

Another aspect to take into consideration is the range of colors the device can display, which is referred to as the color gamut of the display. Color space can be defined as a consequence of the different range of colors, or gamut, a device can handle on a visual plane, such as sRGB, Rec2020, etc. It can be also defined as a tridimensional space used to represent and characterize color, such as CIEXYZ, CIELAB, etc. The organization accepted as the international authority on color, color spaces, on light and illumination is the International Commission on Illumination (CIE for its French name, Commission Internationale de l'éclairage). CIE is recognized by different organizations, such as International Standardization Organization (ISO) or International Commission for Weights and Measures (CIPM), as an international standardization body. Within the CIE's objectives is to develop standards and procedures of metrology in the fields of light and lighting. To prepare and publish standards, reports and other publications related to the fields of light and lighting (CIE).

In 1931, CIE defined the relationship between visible spectrum and the human perception of color as color space. In the same year, the CIE also created two color spaces, CIE 1931 RGB color space and CIE 1931 XYZ color space. Each one with different characteristics with the purpose to standardize how the color spaces are used. Due to their weaknesses, such as having a negative part for the spectrum of the red primary of the CIE RGB, different color spaces have been created like CIELUV in 1976 (CIE).

Nowadays, different color spaces exist which allow us to handle color in different ways. Even though the CIE has worked to standardize procedures in the field of light and lighting, the problem remained for devices be-

cause each one of them has different characteristics and handles different gamut. Making it difficult to match color spaces between devices, such as camera and a printer or a screen. In 1993, the International Color Consortium (ICC) was formed by eight vendors with the purpose of promoting the use and adoption of open, vendor-neutral, cross-platform color management systems.

The primary goal of the ICC was to develop and administer a standard color profile, the founding members committed to support these color profiles in their operating systems, platforms and applications; since then, the consortium has expanded to over 60 members from different industries. These color profiles provide a cross-platform profile format to create and to interpret the color data. They can be used to translate the color data between different color encodings, from one device into another device's native color encoding. For example, a printer company creates a single profile for multiple operating systems and applications.
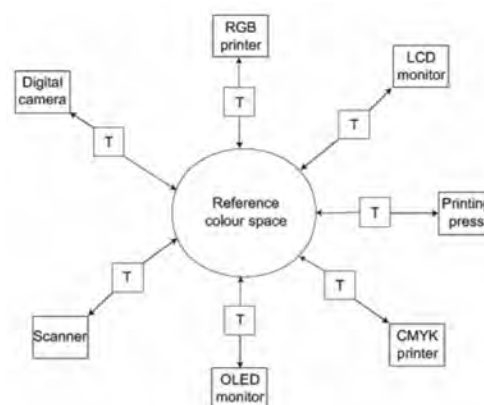
Figure 5: Color management with color profiles (Consortium, 2010).

The workflow of the color profiles is to obtain the color data and translate it into a reference color space, also known as Profile Connection Space (PCS). This workflow allows transformations between different color encodings, used by different devices. For years, the ICC has been improving the specifications needed to create color profiles based on the specifications of the CIE and ISO. As result, there are different versions of the ICC color profiles; the current work is based on the ICC color profile version 4, which is the most widely used today, specifically the revision 4.3 (Consortium, 2010).

The PCS contains in a reference color space all the necessary color data to reproduce the viewing conditions from the device. The reference color, according to the ICC specifications, can be represented using either CIEXYZ or CIELAB; the first one was chosen for this project. It is possible with a sample of these values to define the color appearance for a specified state of viewer adaptation. Due to several standards defined by

the CIE, the ICC had to restrict these options so it could be possible to have an unambiguous color specification system for a particular application (Consortium, 2010).

The problem of the color spaces defined by the ICC lies on the CIE system not being able to contain the information about the illumination or the effect of the surrounds of the sample measured, both affect the appearance. To overcome this problem, the ICC first defined the PCS to be always chromatically adapted to CIE standard illuminant D50, assuming the state of the chromatic adaptation of the viewer. Second, the use of rendering intents which describe the colorimetry of an image (Consortium, 2010).

Chromatic adaptation is a transform which Green and Habib (2019) define it as a method to predict corresponding colors viewed under a different adapting illuminant. In this project, the illuminant source is considered to be D65 which is why the chromatic adaptation must be applied to fulfill the ICC specifications of the PCS to be D50 illuminant. Bradford transform was the method chosen and, according to Green and Habib (2019), little better to CAT16 transform. They also explain the adaptation as a transform from XYZ colorimetry into cone space, performing the adaptation by applying ratios of cone excitations for the source and destination illuminant, and converting back to XYZ.
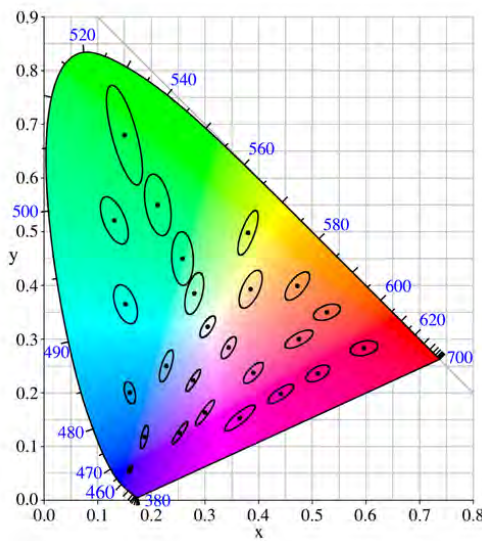
Figure 6: MacAdam ellipses in the CIE 1931 Diagram (Michael Nölle and Boxleitner, 2013).

The ICC specifications version used in this work describes models which perform the transformation between color encoding and can be used according to the user's needs. Each one of the models provides trade-offs in memory, color quality and performance (Consortium, 2010).

The lutBToAType model, Figure 23, is the model with more processing elements, without taking into consideration the multiProcessElementsType model, 5 ele-

ments, which can include an arbitrary number of elements. The fact that the lutBToAType contains more elements than the other models gives more control to process the data and allows to achieve a better performance at the expense of higher memory consumption and calculation complexity.

Badano et al. (2015) refer to color consistency as the ability of the device to produce image data with an identical perceptual response in human response. David MacAdam (1942) was one of the first to determine how the human color perception system works. He conducted experiments on a representative population to define 25 ellipses in which two colors may be considered by an average eye as the same, these ellipses have different sizes and orientations.

Several formulas have been proposed by the CIE over the years to measure the color difference. The CIELAB color space is the color space where the color differences are measured as the Euclidean distance between two samples' coordinates. The CIE DELTA 2000 (CIEDE2000) is the result of iterative improvement of the original Euclidean distance which addresses the "elliptical" perception of color difference, between a sample color and a reference color. It is defined as:

$$\Delta E = \sqrt{\left(\frac{\Delta L'}{K_L S_L}\right)^2 + \left(\frac{\Delta C'}{K_C S_C}\right)^2 + \left(\frac{\Delta H'}{K_H S_H}\right)^2 + R_T \left(\frac{\Delta C'}{K_C S_C}\right)\left(\frac{\Delta H'}{K_H S_H}\right)}$$

(Gaurav Sharma, 2004)
The details of the formula can be found in section 7.

The math defines an ellipsoid around a standard color, the ellipsoid corresponds to the attributes of hue, chroma and lightness. A CIEDE2000 value equal to or less than 1 means the actual color and the standard or reference color difference is not visible to the human eye. A CIEDE2000 value between 2 and 4 means the difference is hardly visible to the human eye. Any CIEDE2000 value higher than 5 is clearly visible to the human eye (Gaurav Sharma, 2004).

The health sector is an area where the color in images is very important too. Especially dermatology which studies the skin, the biggest organ in our body. Nowadays with technology so easy to acquire, dermatologists prefer to have a record of their work by taking pictures of their patients' skin so they can analyze them or have them as evidence before and after a treatment.
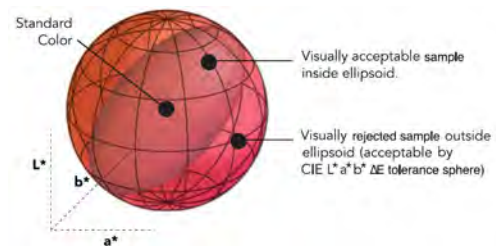
Figure 7: Representation of the "elliptical" perception (X-Rite, 2004).

13.4

One example is the whole-body photographs and detailed close-up pictures of the skin, where one of the things they require is to have a consistent color rendering to avoid erroneous conclusions. Since the purpose of having a digital picture is to be able to see it using any device, such as laptops or displays, it is important to maintain the color consistency through the different devices.

The purpose of this document is to present a pipeline which shows how a color profile helps to maintain the color consistency through different devices, for example from a camera to a display.

2. State of the art

Xi Chen MM and PhD (2020) provides a review of dermoscopy based on published literature; different devices are presented for different diagnostic methods of dermoscopy. Their applications extend from initial differential diagnosis to general dermatology, including nail and hair abnormalities or diseases related to infection and inflammation. Depending on the goal of the physicians is going to be the method and device to use. The devices reviewed are in a wide range of features, ranging from prices to sizes for different diagnostic methods, such as: handheld dermoscopy, viodermoscopy, fluorescence-advanced videodermoscopy, polarized transilluminating dermoscopy and digital dermoscopy.

For digital dermoscopy, Diricx and Kimpe (2019) refer to the importance of having an end-to-end system. The same work quantify color variability originated from different sources. Diricx et al. (2022) present the advantages of leveraging the DICOM standard to enable both standardadization of dermoscopic meta and image data to tackle the increasing need to exchange clinical skin images.
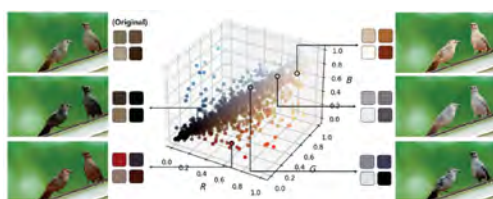
Figure 8: Schematic diagram of interface (Meng-Yao Cui and Lu, 2022).

Artificial Intelligence (AI) has also been implemented on pipelines which allow the users to edit the colors based on their own aesthetics, such as Meng-Yao Cui and Lu (2022). Their research proposes a stage where it learns the color distributions for different objects in the real world by generating color palettes. It also proposes a second stage where the AI recognizes the object category to later recommend realistic candidate colors, the approach uses Mask-RCNN,

Kaiming He and Girshick (2017), trained on the COCO dataset. The last stage is where the interface provides a 3D diagram with color points, they correspond to a palette. The user can change the palette and the object will be recolored, Figure 8 shows the results obtained with different palettes applied to a bird. Even though AI is only used for segmentation, this is an example of how AI can be applied in different stages for different purposes.

The color is an essential component for the pathologists who rely on immunohistochemical stains and colored histochemical to identify structures within the lesion area. As Yagi (2011) described, the major reasons for color variation are the thickness of the tissue, staining, scanner, viewer and display; along with the different protocols and practices in histology labs. Technologies in whole slide imaging (WSI) have been improved in the last years, John Gilbertson and Yagi (2005), giving the professionals the opportunity to use them for several purposes like: remote diagnosis, education, conferences, also to develop artificial intelligence. Due to those situations, the FDA launched for the first time in May 2013 a workshop to discuss color standardization in medical imaging, specifically digital microscopy, endoscopy, medical photography, display and telemedicine, although the discussion remains for different fields.

As Inoue and Yagi (2020) described, there are different methods which try to achieve color standardization. One of the methods mentioned is called color correction in which the conversion of the color space is involved. Another possible method is proposed by Nektarios A Valous and Allen (2009) doing the correction in the linear RGB color space instead of the CIE XYZ color space to perform a CIE color characterization using a computer vision system based on digital photography.

Another method is using a target slide, as the FDA (2016) recommends. The target must contain measurable and representative color patches with the purpose of analyzing the difference and do the correction through a transformation matrix.

A similar method is suggested on the display side, display a color standardization slide and compare between the colors physically and the colors displayed. If any difference, a calibration should be done with a display calibration device.

Society continues to diversify, which is why researches need effective strategies for assessing skin in ways that are socially meaningful. Rachel A. Gordon and Nunez (2022) apart from mentioning the need for new strategies, they examined two most widely used skin tone rating scales (Massey-Martin and PERLA) and two handheld devices. Each one of those scales was created for specific purposes, but through the years they were implemented in other studies because of the variations of skin tones.

Figure 9: PERLA scale (of California, 2008).

For the physical measurement of skin color, the Labby and Nix Mini handheld devices were chosen based on their different features, such as price and their functionality. Labby is a spectrophotometer (captures the full visible spectrum of light), Nix is a colorimeter (captures certain wavelengths).

The goal of Rachel A. Gordon and Nunez (2022) is to know how consistent and how comparable their four measures of skin color are, apart from knowing if the measurements are socially meaningful.

The consistency was examined using the intraclass correlation going from 0.60 to 0.74 for good and 0.75 to 1.0 for excellent. The comparability was examined cross-device comparability for the devices and cross-scale comparability for the scales. The comparability between devices and scales was examined by graphing the average of the scales' ratings against the average CIE Lab values. Statistics were used to evaluate meaningfulness, such as standard deviation, mean and their ratio.

Figure 10: Massey-Martin scale (Massey and Martin, 2003).

The results show excellent consistency between devices, higher than 0.90 intraclass correlation points. The consistency of the scales was also excellent, between 0.83 and 0.91 intraclass correlation points. The comparability between devices were highly linearly related, as well for the scales. Comparability between devices and scales shown a consistently linear association between CIE L values and the scales' values. After analyzing the

results from the statistics to evaluate meaningfulness, they concluded grater variation among black skin tones than white skin tones when classifying the photos.

Medical displays are also evolving to improve image quality through greater efficiency, higher accuracy and more functionality. As the display technology improves, the evaluation of the display gets more complex too. J Penczek and Sriram (2021) focus on the test conditions which these conditions require considering the intended application to get a better performance. Due to the workflow of capturing an image with a device like a digital camera, process it and image rendering, the performance of the displays depends on the image. The OLED displays and LCD displays are an example of technologies which the contrast ratio will vary depending on how much content is rendered on the display. The display industry recognizes the content-dependent performance which is why based on different guidelines, the industry is adopting RGBCMYW multi-color test patterns, as Figure 11 shows.
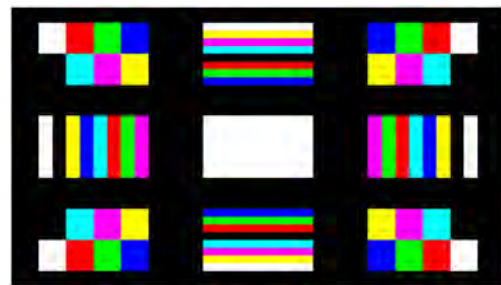
Figure 11: Example of multi-color test pattern (J Penczek and Sriram, 2021).

The color gamut is another condition mentioned by J Penczek and Sriram (2021). sRGB is the most common color space used by devices like cameras and office monitors also used when color accuracy is important, so when the image does not contain information about the color space, it is assumed to be sRGB. However, the sRGB color space is not able to represent many colors from the visible spectrum as the CIE indicated. That is why the color space Rec.2020 or BT.2020 has been recently introduced which covers more colors within the visible spectrum, even though it requires narrow bandwidth light sources.

The last condition mentioned which will affect the chromaticity is the environment where the images are visualized. The chromaticity area shrinks as the ambient illumination increases. This condition will affect especially portable devices such as smartphones and tables.

The DICOM GSDF has been adopted because it ensures consistency only for grayscale images regardless of the device and time. Tom Kimpe and Xthona (2016) proposed an extension of DICOM GSDF, referred as Color Standard Display Function (CSDF), which increases the perceptual linearity of visualized colors.

The algorithm was tested on consumer displays, professional displays and medical grade displays giving good results on images with color and remained compliant with DICOM GSDF standard.

3. Material and methods

Barco NV has a method to detect a ColorChecker Passport and to calibrate an image, so the goal of this project is integrate it into an ICC color profile to make the method widely applicable maintaining the performance obtained with Barco's method.

The color management module chosen for the project was Little CMS because apart from being open source, it uses the ICC standard, providing the necessary tools to encode the data and to connect color profiles. Figure 12 shows an example of the pictures used for the project. The pictures were taken under different conditions to simulate how dermatologists work. The pictures captured a region of skin to analyze and a ColorChecker Passport which is used to calibrate the colors according to the reference values from the ColorChecker.

One important step to ensure end-to-end color consistency is to define to which device the color profile belongs to. Due to this, the ICC defined as a mandatory tags the one called *device class*. The next two subsections, input device and display device, refer to the device assigned for each one of the color profiles.

Figure 12: Example of pictures used.

3.1. Input device

The reason to use the lutBtoAType model is because it allows different combinations to process the color, this must be adjusted to the needs of the user. The second reason is because the 3D LUT was calculated to calibrate the image. For this project, the pipeline used was to set the "A" curves and "B" curves to be identity functions while the Multi-dimensional lookup table, "M" curves and matrix were adjusted. The Profile Connection Space (PCS) was set to CIE XYZ.

Two color spaces were implemented, sRGB and Rec.2020. The first one because, as explained before,

is the color space widely used among display devices. Rec.2020 or BT.2020 was also implemented because it covers a wider space in the visible spectrum so it can be possible to not only cover more colors, but the input devices generating colors outside of the sRGB gamut.
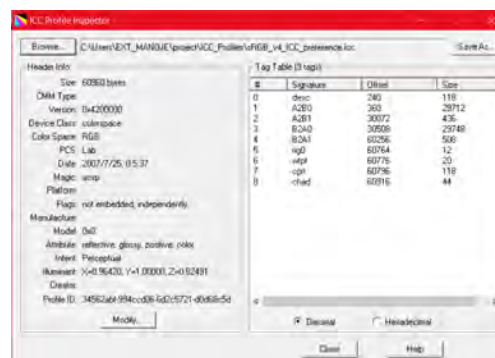
Figure 13: ICC profile inspector interface.

The ICC provides several useful documents, ICC profile files and tools to build, understand and explore ICC profiles which can be found on its website. The ICC profile inspector is one of those tools used in the project too.

The ICC Profile Inspector is free software which opens ICC profiles and makes the content readable. Figure 13 shows the interface of the software showing the information from one of the sRGB color space profiles found in the ICC website.

All the profiles must have a header and a tag table. The content of these two requirements will depend on the choices and needs of the user. The ICC specifies mandatory and optional tags, they depend on the device to profile and the architecture to use.

For an input device, the list shows the mandatory tags with the current values chosen for the project:

- **Profile version**: 4.3

- **Device class**: 'scnr' (Input device)

- **Color space of data**: 'RGB'

- **PCS**: 'XYZ'

- **Rendering intent**: Relative intent

- **Profile creator signature**

- **Profile description Tag**

- **Copyright Tag**

- **Media white point tag**: D50

- **Chromatic adaptation tag**: matrix from RGB to CIE XYZ

- **A to B tag**: this tag should be changed in case the architecture is a different one from the LUT model

Based on Figure 23, three components are adjusted based on the picture and the color space (sRGB or Rec.2020).

"M" curves contain three tone response curves, one per color channel (red, blue and green), the response curves are the same for both color spaces and identical for the three color channels (R, G, B):

$$C_{linear} = \begin{cases} \frac{C_{srgb}}{12.92}, & C_{srgb} \leq 0.04045. \\ \left(\frac{C_{srgb}+0.055}{1.055}\right)^{2.4}, & C_{srgb} > 0.04045. \end{cases} \quad (1)$$

The matrix is the last part of the pipeline which goes from linear RGB to CIE XYZ D50, just as the ICC specifications requests for the PCS. Unlike the "M" curves, matrices are different for each one of the color spaces because they are computed based on the chromaticity coordinates of the color space.

3.2. Display device

Display profiles with the PCS set to CIE XYZ were investigated, but without success. The display profiles found in the ICC website have the PCS set to CIE Lab. Because of that reason, two display profiles were created, both profiles simulate an additive model; one profile is with an sRGB color space, the other is set to Rec.2020.

As before, every stage for the display profiles were created and encoded to the corresponding data with the Color Management Module Little CMS with the following mandatory tags for a display device:

- **Profile version**: 4.3

- **Device class**: 'mntr' (Display device)

- **Color space of data**: 'RGB'

- **PCS**: 'XYZ'

- **Rendering intent**: Perceptual intent

- **Profile creator signature**

- **Profile description Tag**

- **Copyright Tag**

- **Media white point tag**: D50

- **Chromatic adaptation tag**: matrix from RGB to CIE XYZ

- **A to B tag**

- **B to A tag**

The initial approach was to use a software which allows to set the color profile of the images, like GIMP. However, it was not possible to review the values after the PCS with this approach. The goal of having display profiles is to have more control on the workflow using Little CMS, to be able to review all the values after stages on the display side as well.

3.3. Profiles connection

Little CMS provides the necessary tools not only to create, but to connect different profiles, as long as they fulfill the ICC specification 4.3. Little CMS creates transforms between profiles, from a profile to the PCS, or from PCS to a profile.

In order to evaluate in a better way, the workflow presented in this paper, different connections were made as the next list shows, the connections were done for both color spaces, sRGB and Rec.2020:

- **Input profile to PCS**: to evaluate if the profile converts from RGB to CIE XYZ with the chromatic adaptation set to D50.

- **PCS to display profile**: to evaluate if the profile converts from CIE XYZ to RGB with the corresponding chromatic adaptation.

- **Input profile to display profile**: to evaluate the end-to-end workflow.

Figure 24 shows the stages performed to analyze the pipeline. The first two stages were performed on the average values of the patches from the ColorChecker, to calculate the DeltaE 2000 value. The third stage was performed on the average values of the patches. When the results were promising, the whole image was transformed to evaluate the results qualitatively.

The evaluation was performed on every stage mentioned previously, but the results presented in this work will only show the ones from the end-to-end pipeline from the calibration before creating the profile and after connecting the input profile to the display profile. Only those results because the purpose of the project is to confirm the calibration is done with the color profiles and to know how much difference there is between the reference values from the ColorChecker Passport and the average patches' value. If so, to know if the difference is going to be perceptible for the human eye or not based on CIE DeltaE 2000 value.

3.4. Pipeline to generate input ICC profile

Everything was done using C++ in Visual Studio 2019, with different scripts created to have a better control and structure of the code. As Figure 25 shows, there are 5 scripts to control the workflow. The main script only requests from the user the name of the uncalibrated image to process, it also contains two key components:

Chart detection: it acquires the information of the color patches from the ColorChecker Passport

Chart profiler: it runs the corresponding scripts from the diagram to generate the profile

13.8

The *chart profiler* script calls to the *Camera Calibrator* which is in charge of creating the 3D LUT based on the CIE Lab reference data and the RGB observed data. After the 3D LUT is created, the *chart profiler* object sends to the *Input Icc Profile Calculator*:

- 3D LUT

- C++ structure containing the chromaticity from the primary colors, the white and black from the color space in which is being created

- The points of the tone response which the "M" curves contain

- Extra information as name of the file and a boolean to indicate if the profile is going to be saved

The *Input Icc Profile Calculator* will calculate and populate the mandatory and some optional tags. This information will be sent to *Input Icc Profile* who is in charge of encoding all the information according to the ICC specifications with the help of Little CMS. This input profile can be connected with a display profile or can be inspected with the software from the ICC website, ICC Profile Inspector.

4. Results

The results will be presented for 2 color spaces, sRGB and Rec.2020. The presented workflow was applied on several images to confirm performance generalizes properly. Further reporting in this work is limited to four images, for clarity purposes.

The ColorChecker Passport has 24 patches. The DeltaE 2000 values are going to be presented in section 7: 6 neutral patches and 6 color patches (red, green, blue, yellow, magenta and cyan) to train the calibration and 12 color patches used for validation with the corresponding average measured value from all the patches. In this section only the average from all the color patches are going to be presented.

The DeltaE 2000 values were calculated as explained in the section 1 between the reference values from the ColorChecker Passport and the final average values of the patches once they go through the pipeline. All the tables have three rows where the first one, uncalibrated stage, contains the DeltaE 2000 values between the observed patch values and the reference values without applying the color calibration profiles. The second row, Barco's method, contains the DeltaE 2000 values between the reference values and the final observed patch values when only the calibration algorithm is applied. The third row, end-to-end pipeline, contains the DeltaE 2000 values between the reference values and the final observed patch values when the input profile is applied and connected to the display profile.

4.1. sRGB color space

In this subsection, the averages will be presented setting the color space to sRGB for the input profile, along with the original image and the result from connecting the input color profile to the display color profile. The average time to perform the profile connection per picture was ≈2 seconds.

4.1.1. Image A

Stage	Average
Uncalibrated patches	8.05015
Barco's method	1.65525
End-to-End pipeline	1.68369

Table 1: Results from image with sRGB color space.

Figure 14: Original image.

Figure 15: Result after connecting profiles.

4.1.2. Image B

Stage	Average
Uncalibrated patches	3.51231
Barco's method	1.13778
End-to-End pipeline	1.15333

Table 2: Results from image B with sRGB color space.

Figure 16: Original image.

Figure 17: Result after connecting profiles.

4.1.3. Image C

Stage	Average
Uncalibrated patches	4.51846
Barco's method	1.65254
End-to-End pipeline	1.67711

Table 3: Results from image C with sRGB color space.

Figure 18: Original image.

Figure 19: Result after connecting profiles.

4.1.4. Image D

Stage	Average
Uncalibrated patches	6.62284
Barco's method	1.88348
End-to-End pipeline	1.91435

Table 4: Results from image D with sRGB color space.

Figure 20: Original image.
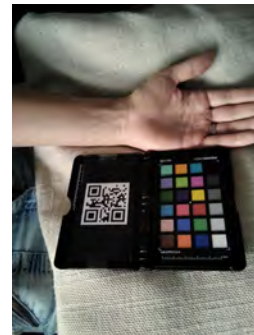
Figure 21: Result after connecting profiles.

4.2. Rec2020 color space

In this subsection, only the averages will be presented without the result of the images because they are in a different color space and they would be seen as if they did not have the necessary quality, this is the consequence of trying to visualize images in a display with a different color space. The average time to perform the profile connection per picture was ≈3 seconds.

4.2.1. Image A

Stage	Average
Uncalibrated patches	8.05015
Barco's method	2.29279
End-to-End pipeline	2.28687

Table 5: Results from image A with Rec.2020 color space.

4.2.2. Image B

Stage	Average
Uncalibrated patches	3.51231
Barco's method	1.99159
End-to-End pipeline	1.99032

Table 6: Results from image B with Rec.2020 color space.

4.2.3. Image C

Stage	Average
Uncalibrated patches	4.51846
Barco's method	2.37119
End-to-End pipeline	2.38938

Table 7: Results from image C with Rec.2020 color space.

4.2.4. Image D

Stage	Average
Uncalibrated patches	6.62284
Barco's method	2.78993
End-to-End pipeline	2.78841

Table 8: Results from image D with Rec.2020 color space.

5. Discussion

Regardless of the color space, the connection between profiles maintain the consistency obtained with the color calibration method provided by Barco. Even though the color spaces and chromaticity change during the profile connection, the results show there is not only color calibration when the color profile is applied but also color consistency which is needed when the images have to be displayed in different devices.

Not only the architecture for the profile can be modified, but the calibration algorithm can be adjusted to the profile builder's needs too. It will depend on what the objective is, but the color will be consistent with the current pipeline through different acquisition devices, display devices or different conditions, such as ambient light.

Qualitatively, the images with the profiles connected have better color quality compared to the uncalibrated image since they are already calibrated, the skin has a better appearance and the details in the skin can be analyzed much closer to the reality in any display. The colors do not seem saturated any more as on the original images.

As future work, it can be implementing the same pipeline with different sizes of ColorCheckers. As the images show, the ColorChecker Passport is big and half

the picture shows the ColorChecker when the goal is to show mainly the skin. Even though the ColorChecker is used once to create the profile and then the ICC profile can be reused with other pictures, a smaller color reference product can be used to make easier the work for the dermatologists and the patients, who hold the object in what may not be a comfortable position.

Another possible future work is to implement the pipeline without any known object in the scene. It may be thought of as relying on the acquisition device automatic white balance and will likely correspond to the uncalibrated image in this work.

IccMax is the newest profile version, 5.0, approved in 2019. It is intended to extend version 4 by providing the benefits of:

- Allow different illuminants for the PCS

- Handles high-precision (32 bit)

- It has five types of inter-profile connections

  - Named
  - Colormetric
  - Spectral
  - Material
  - BRDF

Using the newest version will allow to increase the precision at every stage in the profile generation, reducing the rounding errors presented in the connection of the profiles. These rounding errors are considered to be the main cause of the difference between the *Barco's method* and *End-to-End pipeline* average values in section 4. The use of this version will also help to avoid conversion of illuminants in the PCS which will maintain the original one through all the pipeline.

The reason for not using this version in the current project is due to limited industry adoption. The IccMax CMM is intended to be completely backward-compatible, will recognize and correctly process v2 and v4 profiles. However, IccMAX profiles are not expected to be compatible with v4 CMMs. (Consortium, 2019)
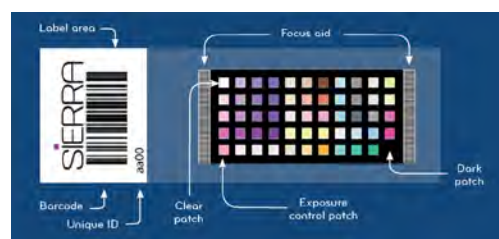
Figure 22: Sierra glass slide (FFEI, 2022).

The method presented in this paper can be applied not only for dermatology, but to other areas within the

health sector, like histopathology. This can be achievable by using a glass slide with the corresponding color patches and their reference values with the purpose of ensuring consistency of color for WSI (Whole Slide Imaging). An example of a glass slide can be the one developed by FFEI: *The Sierra Slide*.

It is a standard size glass slide with 55 biologically stained patches. The importance of knowing the true color from the tissue in the glass slide is so high that there are many models for scanners, each one with different characteristics, without mentioning the amount of displays present in the market to visualize the WSI. Therefore, there must be a bridge between all the variety of visual devices and the color profiles are the ideal tool to keep the consistency of colors. (FFEI, 2022)

One example, histopathology, has been presented of the fields where the workflow of this paper can be implemented, but it does not mean it is the only one. As we have seen, the health sector is expanding and improving very fast, that is why methods like the one presented here or standards should be established so the physicians do not have problems when sharing or analysing the information with different devices.

6. Conclusions

From the clinic to a conference, the physician can be sure the consistency will be kept with the corresponding color profile. Since there is still no standard or a right bridge between different devices, the proposed pipeline can help to share not only the image but also the color. The color is something which is getting more important for the health sector that the institutions, like the International Color Consortium, are taking into consideration its needs to improve its functionality. The algorithm to calibrate the colors in the image can be adjusted to every manufacturer's needs, but the presented workflow to create ICC color profiles will work as long as the devices work with a Color management framework to handle and adjust the color displayed.

Acknowledgements

References

Badano, A., Craig Revie, A.C., Wei-Chung Cheng, P.G., Tom Kimpe, E.K., Christye Sisson, S.S., Darren Treanor, P.B., David Clunie, M.J.F., Tatsuo Heki, S.H., Hiroyuki Homma, A.M., Takashi Matsui, B.N., Masahiro Nishibori, J.P., Thomas Schopf, Y.Y., Yokoi, H., 2015. Consistency and standardization of color in medical imaging: a consensus report. Journal of Digital Imaging , 41–52doi:https://doi.org/10.1007/s10278-014-9721-0.

Barco, 2022. Medical displays. https://www.barco.com/en/products/medical-displays, Last accessed on 2022-06-08.

of California, U., 2008. The Project on Ethnicity and Race in Latin America.

CIE, . International commission on illumination. https://cie.co.at.

CJ Kazilek, K.C., 2009. Colors animals see .

Consortium, I.C., 2010. Specification ICC.1:2010 (Profile version 4.3.0.0).

Consortium, I.C., 2019. Specification ICC.2:2019 (Profile version 5.0.0 - iccMAX.

Dale Purves, George J Augustine, D.F., 2001. Neuroscience, 2nd edition.

Diricx, B., E. De Brauwer, J.D.V., T. Kimpe, L.C., Malvehy, J., 2022. Advantages of standardization of dermoscopic imaging by leveraging the dicom standard .

Diricx, B., Kimpe, T., 2019. Color variability in digital dermoscopy, Presented in the 24th World Congress of Dermatology, Milan, Italy.

Dunlop, J., 2022. Photography for beginners (the ultimate guide) .

FDA, 2016. Technical performance assessment of digital pathology whole slide imaging devices; guidance for industry and food and drug and administration staff.

FFEI, 2022. Sierra slide.

Gandhi, V.C., 2015. Various display technologiess. International Journal of Modern Trends in Engineering and Research .

Gaurav Sharma, Wencheng Wu, E.N.D., 2004. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. Color Research and Application 30. doi:http://doi.org/10.1002/col.20070.

Green, P., Habib, T., 2019. Chromatic adaptation in colour management. CCIW 2019: Computational Color Imaging , 134–144URL: https://doi.org/10.1007/978-3-030-13940-7_11.

Inoue, T., Yagi, Y., 2020. Color standardization and optimization in whole slide imaging. Clin Diagn Pathol doi:http://doi.org/10.15761/cdp.1000139.

J Penczek, P A Boynton, R.B., Sriram, R.D., 2021. Measurement challenges for medical image display devices. Journal of Digital Imaging , 458–472doi:http://doi.org/10.1007/s10278-021-00438-1.

John Gilbertson, A.A.P., Yagi, Y., 2005. Clinical Slide Digitalization: Whole Slide Imaging in Clinical Practice Experience from the University of Pittsburgh. 1st edition ed., CRC Press.

Kaiming He, Georgia Gkioxari, P.D., Girshick, R.B., 2017. Mask r-cnn. 2017 IEEE International Conference on Computer Vision (ICCV) , 2980–2988doi:http://doi.org/10.1109/ICCV.2017.322.

Leelakrishnan, L., 2022. Prism and dispersion of light .

Lindbloom, B.J., 2017. Delta e (cie 2000). https://www.brucelindbloom.com, Last accessed on 2022-06-08.

MacAdam, D.L., 1942. Visual sensitivities to color differences in daylight. Journal of the Optical Society of America 32, 247–274. doi:https://doi.org/10.1364/JOSA.32.000247.

Massey, D.S., Martin, J.A., 2003. The NIS Skin Color Scale.

Meng-Yao Cui, Zhe Zhu, Y.Y., Lu, S.P., 2022. Towards natural object-based image recoloring. Computational Visual Media 8, 317–328. doi:http://doi.org/10.1007/s41095-021-0245-5.

Michael Nölle, M.S., Boxleitner, W., 2013. H2si - a new perceptual colour space doi:http://doi.org/10.1109/ICDSP.2013.6622837.

Nektarios A Valous, Fernando Mendoza, D.W.S., Allen, P., 2009. Colour calibration of a laboratory computer vision systeme for quality evaluation of pre-sliced hams. Meat Sci doi:http://doi.org/10.1016/j.meatsci.2008.07.009.

Paul, A., 2022. What are the different types of medical imaging equipment? https://www.wisegeek.net/what-are-the-different-types-of-medical-imaging

-equipment.htm, Last accessed on 2022-06-08.

Plaza, C., 2013. Science fridays: A tale of two models.

R. Sharan, K. R. Sarma, B.M., Iyer, S.S.K., . Display Technologies (ICT and Visualization).

Rachel A. Gordon, Amelia R. Branigan, M.A.K., Nunez, J.G., 2022. Measuring skin color: consistency, comparability, and meaningfulness of rating scale scores and handheld device readings. Journal of Survey Statistics and Methodology , 337–364doi:http://doi.org/10.1093/jssam/smab046.

Rick, D., 2013. A brief history of light and photography .

Sack, H., 2016. James clerk maxwell and the very first colour photograph .

Sliney, D.H., 2016. What is light? the visible spectrum and beyond doi:http://doi.org/10.1038/eye.2015.252.

Tom Kimpe, Johan Rostang, G.V.H., Xthona, A., 2016. Color standard display function: A proprosed extension of dicom gsdf. Med Phys doi:http://dx.doi.org/10.1118/1.4959544.

X-Rite, 2004. The Color Guide and Glossary.

Xi Chen MM, Quansheng Lu MM, C.C.M., PhD, G.J., 2020. Recent developments in dermoscopy for dermatology. Journal of Cormetic Dermatology doi:http://doi.org/10.1111/jocd.13846.

Yagi, Y., 2011. Color standardization and optimization in whole slide imaging doi:http://doi.org/10.1186/1746-1596-6-S1-S15.

7. Annexes

7.1. Annex A
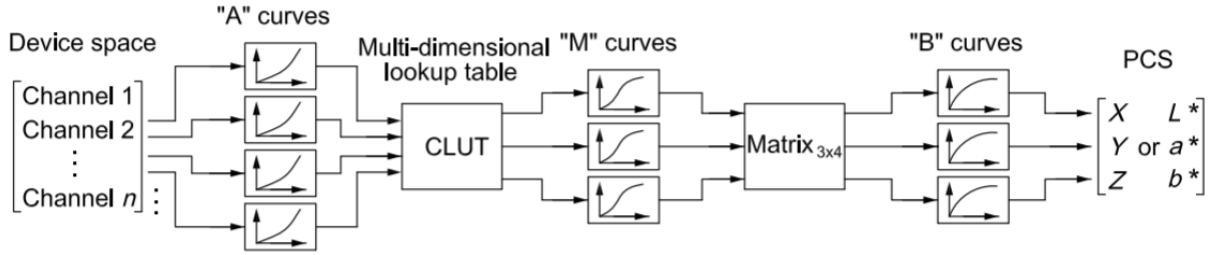
Figure 23: lutBToAType Model (Consortium, 2010).

7.2. Annex B

$$\Delta E = \sqrt{\left(\frac{\Delta L'}{K_L S_L}\right)^2 + \left(\frac{\Delta C'}{K_C S_C}\right)^2 + \left(\frac{\Delta H'}{K_H S_H}\right)^2 + R_T \left(\frac{\Delta C'}{K_C S_C}\right)\left(\frac{\Delta H'}{K_H S_H}\right)}$$

Where:

$$\bar{L}' = (L_1 + L_2)/2$$

$$C_1 = \sqrt{a_1^2 + b_1^2}$$

$$C_2 = \sqrt{a_2^2 + b_2^2}$$

$$\bar{C} = (C_1 + C_2)/2$$

$$G = \frac{1}{2}\left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right)$$

$$a_1' = a_1(1 + G)$$

$$a_2' = a_2(1 + G)$$

$$C_1' = \sqrt{a_1'^2 + b_1^2}$$

$$C_2' = \sqrt{a_2'^2 + b_2^2}$$

$$\bar{C}' = (C_1' + C_2')/2$$

$$h_1' = \begin{cases} \arctan(b_1/a_1'), & \text{if } \arctan(b_1/a_1') \geq 0. \\ \arctan(b_1/a_1') + 360°, & \text{otherwise.} \end{cases}$$

$$h_2' = \begin{cases} \arctan(b_2/a_2'), & \text{if } \arctan(b_2/a_2') \geq 0. \\ \arctan(b_2/a_2') + 360°, & \text{otherwise.} \end{cases}$$

$$\bar{H}' = \begin{cases} (h_1' + h_2' + 360°)/2, & \text{if } |h_1' - h_2'| > 180°. \\ (h_1' + h_2')/2, & \text{otherwise.} \end{cases}$$

$$T = 1 - 0.17\cos(\bar{H}' - 30°) + 0.24\cos(2\bar{H}') + 0.32\cos(3\bar{H}' + 6°) - 0.20\cos(4\bar{H}' - 63°)$$

$$\Delta h' = \begin{cases} h'_2 - h'_1, & \text{if } |h'_1 - h'_2| \le 180°. \\ h'_2 - h'_1 + 360°, & \text{else if } |h'_1 - h'_2| > 180° \text{ and } h'_2 \le h'_1. \\ h'_2 - h'_1 - 360°, & \text{otherwise.} \end{cases}$$

$$\Delta L' = L_2 - L_1$$

$$\Delta C' = C'_2 - C'_1$$

$$\Delta H' = 2\sqrt{C'_1 C'_2}\sin(\Delta h'/2)$$

$$S_L = 1 + \frac{0.015(L'-50)^2}{\sqrt{20+(L'+50)^2}}$$

$$S_C = 1 + 0.045\bar{C}'$$

$$S_H = 1 + 0.015\bar{C}'T$$

$$\Delta\theta = 30\exp\left\{-\left(\frac{\bar{H}'-275°}{25}\right)^2\right\}$$

$$R_C = 2\sqrt{\frac{\bar{C}'^7}{\bar{C}'^7+25^7}}$$

$$R_T = -R_C\sin(2\Delta\theta)$$

$$K_L = 1\,default$$

$$K_C = 1\,default$$

$$K_H = 1\,default$$

Details of CIE Delta formula (Lindbloom, 2017).

7.3. Annex C
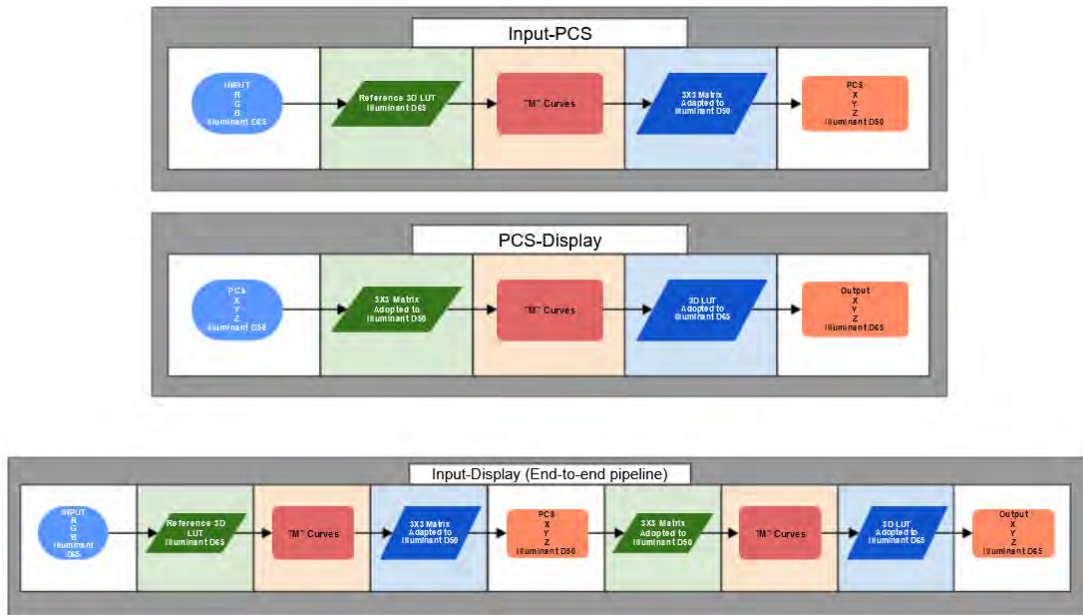
Figure 24: Diagram of how the different stages were evaluated.

13.15

7.4. Annex D
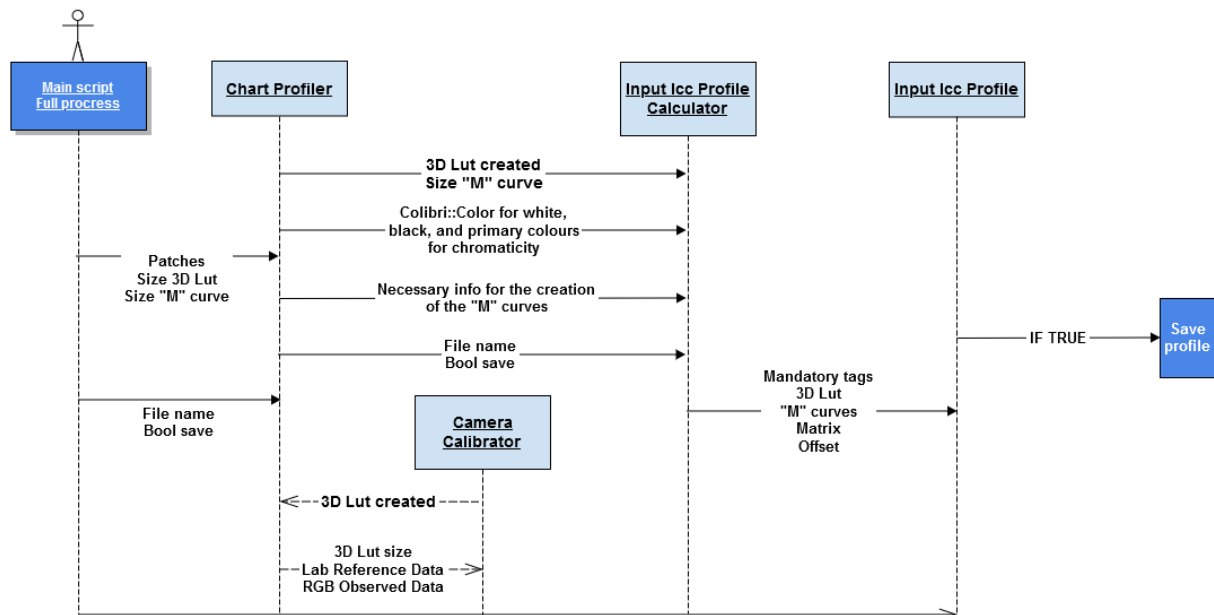
Figure 25: Diagram with the pipeline to create the color profile.

7.5. Annex E

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	8.70154	7.78226	8.41445	8.34829	10.5412	6.7749	4.56056	7.90138	5.76417	11.8233	7.38265	5.15672	6.26027	6.42148	4.99669	2.9518	7.39478	5.41274	15.2102	13.8557	13.692	12.6431	9.50946	1.70401	8.05015
Barco's method	2.71535	3.48359	2.51719	3.20443	0.49417	3.76243	4.21906	1.33278	0.94889	5.64657	3.42101	5.2639	0.0150823	0.0512597	0.0238986	0.0115413	0.0019211	0.26204	0.297645	0.275444	0.259029	0.294019	0.921833	0.348985	1.65525
End-to-End pipeline	2.77783	3.49135	2.53726	3.29833	0.493135	3.74505	4.22158	1.38042	0.957288	5.6942	3.44221	5.24565	0.10692	0.032416	0.0355125	0.00543299	0.00556905	0.239908	0.320579	0.305875	0.29822	0.328137	1.06319	0.382554	1.68369

Table 9: Results from image A with sRGB color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	3.6065	2.85934	5.13302	3.95097	0.593825	4.05146	4.58237	4.80082	2.9308l	2.76783	4.11269	4.55663	2.51038	5.14126	3.72288	3.7458	3.97078	1.88331	3.93191	1.81262	2.95806	3.8822	4.37932	2.40156	3.51231
Barco's method	2.93842	3.18558	3.64305	0.31691	2.40866	0.56578	4.5137	3.1216	0.483119	2.25779	1.16143	1.89989	0.0021549	0.0098121	0.0118291	0.0373634	0.0208938	0.0210201	0.201159	0.184068	0.0256176	0.142616	0.148362	0.00593402	1.13778
End-to-End pipeline	2.93495	3.18731	3.6664	0.318773	2.88995	0.599251	4.49543	3.19594	0.480061	2.26781	1.16321	1.89485	0.126871	0.0544088	0.070046	0.0275796	0.0199697	0.0379323	0.197775	0.162334	0.0592587	0.133228	0.135746	0.113894	1.15333

Table 10: Results from image B with sRGB color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	11.283	3.78565	5.55156	6.36561	4.22087	5.33859	6.31367	4.26144	5.27407	7.07224	5.00587	2.98843	2.75437	8.40992	6.00433	4.41111	2.87893	2.20642	2.14505	0.832361	1.49334	3.47427	5.18966	1.19024	4.51846
Barco's method	8.42645	3.22838	3.96161	1.28143	3.25957	2.96698	4.93227	1.9294	0.399551	5.2611	2.0706	0.777022	0.00716534	0.000696097	0.00401145	0.00983824	0.0016551	0.00110329	0.227189	0.0753845	0.42465	0.170429	0.145857	0.0624504	1.65254
End-to-End pipeline	8.45595	3.23339	4.00139	1.32365	3.29014	3.00858	4.93153	1.93967	0.396425	5.27414	2.08819	0.776711	0.0933349	0.050243	0.0197486	0.0128619	0.00111875	0.176407	0.234033	0.0980827	0.408775	0.205018	0.184573	0.0666044	1.67711

Table 11: Results from image C with sRGB color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	9.04955	5.76964	7.27925	7.11691	7.22259	5.19414	6.91179	7.91493	8.46311	7.7026	6.21447	8.51249	6.34732	6.92016	8.63388	4.23058	6.66518	7.45561	8.1576	7.1921	6.82475	6.43995	2.63273	0.0998742	6.62284
Barco's method	5.20981	5.02578	4.69621	3.07811	0.636983	2.00462	7.98461	2.01409	0.87475	2.68916	2.67888	6.82441	0.00887733	0.00558569	0.00232011	0.014672	0.00458472	0.00738845	0.141248	0.183211	0.200291	0.515933	0.247791	0.152377	1.88348
End-to-End pipeline	5.22515	5.02842	4.70745	3.10092	0.627839	1.98657	7.96995	2.07032	0.882063	2.7482	2.70719	6.81276	0.134099	0.0365286	0.0151776	0.00240802	0.00358825	0.222245	0.160279	0.211063	0.232267	0.57018	0.335157	0.154616	1.91435

Table 12: Results from image D with sRGB color space.

13.17

7.6. Annex F

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	8.70154	7.7826	8.41445	8.34829	10.5412	6.7749	4.56056	7.90138	5.76417	11.8233	7.38265	5.15672	6.25027	6.42148	4.99669	2.9518	7.39478	5.41274	15.2102	13.8557	12.6431	9.59046	4.37932	1.70401	8.05015
Barco's method	1.38509	7.15231	1.56417	2.82559	1.01588	2.00075	9.89639	1.07575	4.41854	6.46394	2.84795	11.6004	0.0154155	0.0194728	0.0468405	0.0154288	0.301544	0.218775	0.224954	0.371482	0.402237	0.847275	0.189907	0.287094	2.29279
End-to-End pipeline	1.36944	7.15368	1.56125	2.83421	1.02187	1.99902	9.92082	1.07836	4.41833	6.46893	2.84382	11.6064	0.0129019	0.0185098	0.0318646	0.0515085	0.01812	0.226199	0.296387	0.219508	0.363129	0.394569	0.830752	0.15526	2.38687

Table 13: Results from image A with Rec.2020 color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	3.6065	2.85934	5.13302	3.95097	0.593825	4.05146	4.58237	4.80082	2.93981	2.76783	4.11269	4.55663	2.51038	5.14126	3.72288	3.7458	3.97078	1.88331	3.93191	1.81262	2.95806	3.8822	4.37952	2.40156	3.51231
Barco's method	5.80103	4.817	4.67232	0.601535	3.88681	4.61318	6.79568	4.77217	1.59275	3.64304	0.669387	4.20343	0.00731696	0.0107763	0.0521387	0.0480708	0.123564	0.128564	0.446073	0.204951	0.174826	0.282217	0.18907	1.43188	1.99159
End-to-End pipeline	5.7949	4.81066	4.66511	0.607295	3.89196	4.60216	6.80675	4.75803	1.58845	3.64563	0.66851	4.20824	0.0134246	0.0113724	0.0515919	0.0619847	0.0430775	0.0930568	0.449467	0.205488	0.176605	0.278262	0.214439	0.131142	1.99032

Table 14: Results from image B with Rec.2020 color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	11.283	3.78665	5.55156	6.35661	4.22087	5.53859	6.31567	4.26144	5.27407	7.07224	5.06587	2.98843	2.75437	6.06433	4.41111	2.87893	2.20642	2.14505	0.832461	1.49334	3.47427	5.18946	1.19024		4.51846
Barco's method	10.7561	5.44615	4.79374	0.936282	4.08693	6.84313	7.4841	3.24274	1.8541	6.85377	1.29863	1.65315	0.0180306	0.00686259	0.0281189	0.00144754	0.00638729	0.0601231	0.269128	0.165874	0.044166	0.209912	0.218494	1.30262	2.37119
End-to-End pipeline	10.7554	5.4456	4.78839	0.930952	4.03591	7.37264	7.49206	3.23903	1.85093	6.85408	1.30024	1.65976	0.0188936	0.0045227	0.026447	0.0188936	0.00653111	0.0860741	0.269448	0.157292	0.665332	0.204026	0.186233	0.0711522	2.38938

Table 15: Results from image C with Rec.2020 color space.

Stage / Patch	dark skin	light skin	blue sky	foliage	blue flower	bluish flower	orange	purplish blue	moderate red	purple	yellow green	orange yellow	blue	green	red	yellow	magenta	cyan	white	light gray	light medium gray	medium gray	dark gray	black	average
Uncalibrated patches	9.04955	5.76964	7.27925	7.11691	7.22239	5.19414	6.91179	7.91493	8.46311	7.7026	6.21447	8.51249	6.34732	6.92016	8.63388	4.23058	7.45461	6.66318	7.43461	8.1576	7.1921	6.82475	6.43995	2.62273	6.62384
Barco's method	7.96896	8.72762	4.82272	1.20535	1.20116	3.54323	12.5139	1.75973	3.60965	4.56413	2.15992	11.6743	0.0247036	0.00798977	0.00231797	0.000628	0.0105252	0.0121231	0.189022	0.0898994	0.171721	0.452543	0.102317	0.0542909	2.78993
End-to-End pipeline	7.98628	8.7246	4.80932	3.20969	1.19906	3.5914	12.5185	1.76198	3.60735	4.58124	2.15451	11.6831	0.0242555	0.00515942	0.0025911	0.0095131	0.0124781	0.189862	0.185962	0.0902432	0.162532	0.433028	0.0963564	0.032476	2.78841

Table 16: Results from image D with Rec.2020 color space.
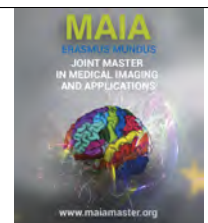
Fusion strategies for multi-modal left ventricle segmentation

Cylia Ouadah

cylia.ouadah@gmail.com

Alain Lalande

Alain.Lalande@u-bourgogne.fr

Sarah Leclerc

Sarah.Leclerc@u-bourgogne.fr

Abstract

Delayed enhancement magnetic resonance imaging (DE MRI) is particularly useful to evaluate the state of the heart after myocardial infarction (MI). To measure the relative extent of MI and helps assess the myocardium tissue viability, automatic segmentation of the myocardial border is necessary. In the last decade, Deep learning methods have reached quite good results for medical image segmentation, however, more precision and robustness are still required for future practical clinical use. In this work, we focus on the use of combined information from both kinetic MRI (CINE) and delayed enhancement MRI (DE) modalities for left ventricle segmentation, and its impact on DE myocardium segmentation. In this study, we introduce a newly constructed dataset CINEDE, that contains MRI volumes of 124 patients for both modalities. Different multi-modal fusion strategies are presented that we can summarize in three categories : early fusion, late fusion and intermediate fusion. In total, five different strategies are investigated whose architectures are all U-Net based. Image registration of CINE images to DE is introduced to study its impact on the results. Furthermore, heart localization using Mask R-CNN is used to guide the registration process towards the structure of interest. The results show that registration helps improving the the segmentation in multi-modal fusion.

In comparison with single modality segmentation, the intermediate fusion architectures, particularly DualUNet, seem to be more robust and more precise for the myocardium segmentation on the test set, as it obtained a Dice score of 0.81 in compared to 0.77 for single modality. Furthermore, the fusion based models have the advantage of providing good results on the CINE modality, which gives additional information to help the heart viability evaluation. On the other hand, simple fusion schemes did not reach the performance of single modality. Preliminary results using ROI detection before the segmentation indicate that multi-modal fusion potentially helps in the localization of the left ventricle.

Keywords: Deep multi-modal segmentation, fusion strategies, myocardium infarction, MRI, CINE, DE, registration, localization.

1. Introduction

Myocardial infarction (MI) is the most common manifestation of ischemic heart disease (IHD). Colloquially called "heart attack", it can be defined as the death of the myocardial cell secondary to prolonged lack of oxygen supply (ischemia). Prognosis estimation after MI relies heavily on the evaluation of the considered segment from cardiac MRI.

Magnetic resonance imaging (MRI) is the most frequently used modality for non-invasive cardiac imaging assessment.MRI provides information about the cardiovascular system structure and function, for this purpose multiple acquisition techniques exist. In the presented

study two of them are used :

- Kinetic magnetic resonance denoted as CINE-MRI generally used to capture motion, wall thickening and volumes of the left ventricle in diastole and systole.

- Delayed enhancement MRI denoted as DE-MRI, that is also known under the name of late gadolinium enhancement (LGE) as the acquisition is based on the use of gadolinium contrast agents which captures abnormal myocardial regions related to the difference uptake of the agent. On these images, the normal myocardium appears dark and the diseased area bright.

The analysis of MRI images helps in diagnosis, treatment and monitoring of several heart conditions including myocardial infarction. Cardiac segmentation is considered as one of the most important methods to this end, since it allows to have medically interpretable measurements. The anatomical structures of interest for this task are typically the heart chambers: left ventricle (LV), the right ventricle (RV), the left atrium (LA) and the right atrium (RA). Because of its greater function and medical interest, most of the cardiac segmentation work in the literature is focused on the left ventricle. The main parameters for evaluation of the cardiac viability are the wall thickness and thickening from CINE MRI and the extent of abnormal area from DE MRI. Obtaining these parameters requires the knowledge of the myocardial contours.

The standard segmentation of the LV consists in defining two boundaries and two regions, the endocardium boundary that separates the cavity from the myocardium, and the epicardium that separates the myocardium from the surrounding tissues. This segmentation allows to compute cardiac parameters like systolic and diastolic volumes, ejection fraction, wall thickness and thickening and myocardium mass when using the CINE modality (Bernard et al., 2018). Most importantly, it allows the quantification of the MI extension using DE modality (Lalande et al., 2022). The local and global study of the cardiac function could inform on the severity of the condition and the potential recovery of the myocardium. As stated, delayed enhancement MRI is generally used for the assessment of the myocardial segment viability and is the standard reference for the detection of myocardial infarction (MI) (KIM et al., 1999). Accordingly, the precision of the myocardial borders detection is a determining factor to estimate the extension of MI. Challenges like EMIDEC (automatic Evaluation of Myocardial Infarction from Delayed-Enhancement Cardiac MRI, http://emidec.com/) showcased the potential of automatic methods on DE MRI in clinical use for prognosis evaluation after a myocardial infarction. Despite the promising results, performance and robustness improvements are needed to have clinically reliable automatic models. On the other hand, automatic methods work efficiently on CINE MRI modality for the myocardium segmentation (Bernard et al., 2018), due to a better contrast between the cavity, myocardium and surrounding tissues. This work will study the possibility of segmentation improvement on DE MRI using the information of CINE and DE images obtained during the same exam.

Compared to a previous study made on this topic (Hadadi, 2021), this work will focus on several intermediate fusion strategies that can be applied when using two image modalities in addition to input and output fusions. Comparing the outcome with the single modality performance will allow to evaluate how much the CINE information could be useful in the segmentation task of DE MRI.

Indeed, we know that both CINE and DE segmentation are complementary in providing helpful measurements for pathology diagnosis or treatment planning by radiologists. Specifically, for MI prognosis, evaluation of the CINE images is used for heart volume, accurate delineation, and contraction information while the one of DE images is used for quantification of the injured myocardium. Therefore, we explore a multi-task approach in this work, in order to obtain segmentations for each modality at once, to benefit from the complementary information.

2. Literature Review

2.1. DE segmentation

In magnetic resonance imaging, left ventricle segmentation faces a lot of challenges such as poor contrast between the myocardium and the surrounding structures, gray level inhomogeneities, relative poor resolution, partial volume effect and other misleading artefacts. However, manual delineation of cardiac MRI performed by experts remains the reference for clinical applications, it is especially time consuming as it takes up to 20 minutes for CINE images (half this time for DE images) to draw the boundaries on one volume (C. Petit-jean, 2011). In addition, manual annotation is sensitive to inter- and intra- observer variability. Thus, in recent years the automation of cardiac segmentation has become of great research interest and where deep learning based methods have reached considerably high performances in cardiac segmentation tasks. This section will focus more on automatic segmentation for DE modality as it allows not only to delineate the myocardium but also the extension of a potential infarction. Still, it is important to mention that the CINE MRI segmentation was widely addressed in this field (Bernard et al., 2018).

A first international challenge organized in 2012 was dedicated for DE segmentation (R. Karim et al., 2016),

during which several automatic and semi-automatic traditional image processing and machine learning methods were proposed. Later with the EMIDEC challenge (Lalande et al., 2022), more up-to-date automatic methods were introduced to the task, relying mainly on deep neural networks.

In medical imaging, the most frequently used deep learning models for semantic segmentation are encoder-decoder based architectures. The reference model for this task is the U-Net introduced in 2015 (Ronneberger et al., 2015). Several novel architectures that were introduced in DE segmentation challenges are based on the generic U-Net architecture, introducing additional blocks or other slight modifications. For instance, attention blocks (Oktay et al., 2018) are now commonly used in medical imaging segmentation to focus on the most significant features at the skip connections stage. (K. Brahim et al., 2021) proposed a novel architecture for the myocardial segmentation on DE images that incorporates attention modules into the U-Net architecture. Other works constructed models that modify the reference encoder and decoder blocks of a U-Net. (Hu J., 2018) applied squeeze and excitation block (SE) at the encoder part replacing the usual convolution-pooling architecture. He et al. (K. He et al., 2016) on the other hand tried to integrate residual blocks into the architecture. According to the latest review for the EMIDEC challenge (Lalande et al., 2022), the best performance for the myocardium and the infarction delineation was accomplished by the model introduced in the work of (Y. Zhang et al., 2021). This model consists of a cascade of two U-Nets, a 2D U-Net that gives coarse results followed by a 3D U-Net that takes into account the original volume and the output of the first U-Net at the input stage. This latter model reached an average Dice score of 87,9% and 13.01 mm for Hausdorff distance which remains unsatisfactory for a clinical application.

2.2. Deep multi-modal scene segmentation

In high level computer vision tasks such as segmentation, deep learning methods have reached excellent performances. Nevertheless, automatic methods still struggle with complex environments. Hence, multi-modal deep learning was introduced to help increase the segmentation performance especially using data from multiple sensors that enables the combination of different information from the exact same scene, making for more robust models. (Y.Zhang et al., 2021) have summarized in their work the different deep learning based fusion approaches in scene segmentation using different datasets and modality (RGB-Depth, RGB-NIR etc). The first attempt of deep multi-modal fusion was a simple input fusion that consists of concatenation into multiple channels (Couprie et al., 2013), this fusion approach is generally classified as an early fusion approach as the fusion of the modalities is prior to feeding the network. Another attempt of fusion was introduced

with FuseNet (Hazirbas et al., 2017) for RGB-Depth semantic scene segmentation. The key idea of this recent approach is to fuse simultaneously at each resolution stage the features maps obtained from the latent space of the encoder into the encoder-decoder network segmenting the RGB. FuseNet motivated the apparition of several new fusion approaches including RFBNet (Deng et al., 2019) that introduces a residual fusion unit block, which consists of two modality-specific residual units and one gated fusion for an interactive fusion exploring interdependence between the encoders' information. Another fusion category is the late fusion that consists of concatenating the information at a late stage of the training process or at the end of it. For example, (Cheng et al., 2017) proposed a gated module fusion to adaptively merge Depth and RGB score maps according to their weights contribution. Another late fusion approach is introduced in (A. Valada, 2017), where feature maps are extracted separately from both modalities and then before the computed maps are summed up for joint representation, followed by a series of convolutional layers to give a final prediction.

In the latter presented works, while the strategies helped the segmentation performance, they might not satisfy the requirements of robustness and accuracy. Thus, other strategies were explored like intermediate or hybrid fusion networks. Some of these approaches combined early and late fusion perspectives, for instance (Guo et al., 2019) presented a network combining a fully convolutional neural network of RGB-D (DFCN) and a depth-sensitive fully-connected conditional random field (DCRF). They stated that the DFCN module can be considered as an extension of FuseNet while the DCRF module is used to refine the preliminary prediction (Y.Zhang et al., 2021). Another recent work on multi-modal fusion is the model presented in (Valada et al., 2019), tackling the problem of fusion by introducing a self supervised modal adaptation (SSMA) module. It dynamically adapts the fusion of multi-scale representations. The network is composed of a ResNet-50 based encoder and an efficient atrous spatial pyramid (eASPP) module that links the encoder to the decoder. Its advantage is to further learn multi-scale features and capture long range context. This latter approach performed the best on Cityscapes dataset and ranks in top 5 in other databases. It is therefore considered as the state of the art fusion method in multi-modal scene semantic segmentation (Y.Zhang et al., 2021). The conclusion that we can make from this bibliographic research is that multi-modal images captured from different sensors give complementary information for the segmentation task, and while there are several fusion strategies it is hard to conclude about an optimal fusion since the results differ depending on the dataset and that the experiments are not always comparable (Y.Zhang et al., 2021).

2.3. Deep multi-modal segmentation in medical imaging

With the development of medical imaging acquisition systems, multi-modal image analysis has been significantly growing for a number of applications. One of these applications is automatic segmentation. We can find several works in the literature applying complementary modalities from different systems (for instance : Computed tomography (CT) and positron emission tomography (PET)) or from the same imaging device, (ex: T1, T2 weighted images in MRI) for the segmentation task. (Guo et al., 2019)'s work demonstrates the superior performance of using multi-modal information including MRI, CT scans and PET images for the segmentation of lesions in soft tissue sarcomas compared to the same architectures trained on a single modality. This work also concluded that the fusion at the layers stage performed better than the output fusion strategy. (Zhou et al., 2019) published a review paper on deep learning for segmentation using multi-modal images. The work investigated DenseNet architectures and Generative Adversarial Networks (GAN) based strategies among others, and three fusion approaches : input fusion, layer level fusion and output fusion. The experiments were made using different combinations of modalities fusion between MRI T1, T2, Flair and CT for brain imaging. The comparison of the different fusion strategies highlighted that the output fusion worked better than the input fusion, which can be explained by the fact that this technique trains two networks experts on the two modalities,which requires however more computational resources. On the other hand, the layer level DenseNet based fusion approach outperformed the other strategies as the network can better learn the complex relationships between the two modalities used, thanks to the dense connections among the layers (Zhou et al., 2019). From previous studies, the main problems in multi-modal segmentation in medical imaging remains the lack of data and lack of generalization ability (i.e. robustness when segmenting unseen data). These tasks along with the complex deep neural networks require large amounts of data so that the inquiry of the fusion choice would be resolved more efficiently, conducting to reliable results.

In the work of (Hadadi, 2021), where the dataset used was limited to 76 patients in total, experiments of CINE and DE segmentations were carried out for input and output fusion. Results using cross validation showed that multi-modal segmentation after registration helps the myocardium segmentation on DE MRI. Additionally, the work concluded that output fusion tends to give better results than the input fusion. It is also necessary to mention that in this study, due to the small amount of data, the results achieved with single modality segmentation were not comparable to the results found in the literature.

3. Material and methods

3.1. CINEDE Dataset

The CINEDE dataset was built from a previous dataset composed of the exams from 76 patients (Hadadi, 2021), which was enlarged along this work up to 124 patients. Images are acquired at the University Hospital of Dijon, using two MRI scanners of different magnetic field strength (1.5 T - Siemens Area and 3.0 T - Siemens Trio Tim, Siemens Medical Solutions, Germany). The name CINEDE refers to the two MRI modalities CINE and DE, as it contains 124 CINE MRI volumes and their 124 corresponding DE MRI volumes, the ground truth (i.e. the expert annotations) is also provided for both modalities separately. Each MRI volume is composed of several slices (6 to 11) acquired from a short axis view of the left ventricle from the base to the apex of it. Short axis view is used for acquisition as it gives an excellent cross sectional view of the left ventricle of the heart (see Figure 1).
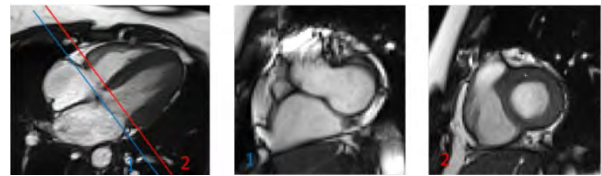
Figure 1: Short axis view MRI acquisition - two slices example (Marchesseau et al., 2016)

The resolution of the MR images varies between and within modalities. For the CINE modality, the range of the image resolution starts from 1.21 (mm) to 1.64 (mm), with a mean value of 1.38 (mm) and a median value of 1.37 (mm). Meanwhile, the resolution for the DE modality ranges from 1.45 (mm) to 2.19 (mm), with a mean value of 1.82 (mm) and a median value of 1.87 (mm). All in all, DE images have poorer resolution compared to CINE images. This resolution will be harmonized between and within the modality as a preprocessing step.

In total, number of slices in the CINEDE dataset is 984 2D slices for each modality. It has the particularity of being very heterogeneous, because the volumes are from different patients, a number of them suffer from several cardiac pathologies. The cases can be grouped in the following categories :

- Myocarditis : it consists of the inflammation of the heart muscle.

- Dilated cardiomyopathy : it is a disease that causes the heart chambers to thin and stretch, growing larger.

- Hyperthrophic cardiomyopathy : The walls of the heart chambers become thicker.

- Myocardial infarction : It happens when blood flow to the coronary arteries of the heart decreases or stops.

- Other pathologies : They are different rare pathologies such as amyloidosis, or Tako Tsubo syndrom ... with a modification of the heart shape. Their low presence in the dataset is the reason they are grouped in one category.

- Normal exams.

In CINE and DE MRI, the left ventricle cavity have higher gray levels than the surrounding myocardium (with DE images brighter than CINE images). Since DE MRI is acquired 10 minutes after the injection of the contrast agent (gadolinium based), the traces of this contrast agent are visible only in pathological zones. Therefore, unlike in CINE MRI, MI is translated by a hyper signal in the DE modality (see Figure 2). This tends to make myocardium segmentation a harder task, compared to the other cases where the intensities are more homogeneous along the myocardium.
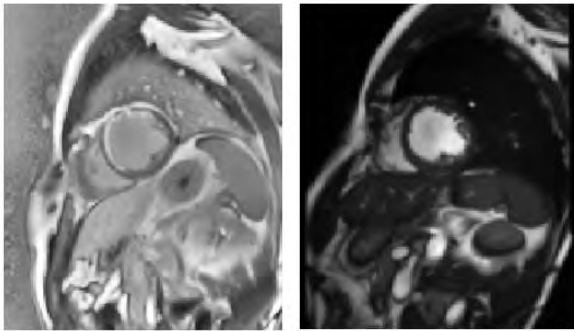
Figure 2: Myocardial infarction in DE VS in CINE modalities

3.2. Software and Hardware details

This project was developed using Python 3.9.1, CUDA 11.4 and PyCharm 2021.3.2 IDE. To read and manipulate NIFTI data, the nibabel 3.2.2 library was used. For image processing tasks, we remained on OpenCV 4.5.5 and Pillow (PIL) 9.0.1. SimpleITK 2.0.2, a registration and segmentation toolkit available in Python, was used for image registration in the pre-processing part. Medical Open Network for Artificial Intelligence (MONAI : https://monai.io/) provided ready to use loss functions for the segmentation task.

The networks were trained on one of the 4 Nvidia DGX GPUs with 32 GB of builit-in RAM each, pro-vided by Mésocentre de calcul, Besançon.

3.3. Preprocessing

In deep multi-modal networks segmentation, con-tradictory information from complementary modalities

may affect the model performance considerably. As mentioned in the presentation of the CINEDE dataset, delayed enhancement images are not acquired at the same time as CINE images, and since the acquisition are made in apnea, the heart localization often differ from one apnea to another. Thus, the views may not match in the position of the heart, and even in the con-tent since one modality might contain more of the sur-rounding or heart tissues than the other one. To help tackle this problem and other usual image analysis prob-lems, some pre-processing steps were introduced before the training phase that will be elaborate point by point in this section.

Before introducing the main preprocessing steps, some initial modifications made on the dataset are worth mentioning :

- Data normalization : The intensities of the CINE and DE images were normalized to range be-tween 0 and 255, using the min-max normalization mehtod.

- Reorientation : The orientation of CINE and DE of the same case happens to be too different , there-fore manual reorientation of one of the modality in some cases was needed.

- Cleaning : The dataset was cleaned from the slices that were not provided with the ground truth in one or both modalities, resulting in 984 2D slices after this step.

3.3.1. Image resolution adjustment and zero-padding

The two modalities have different voxel spacing, however it is important to adjust the spatial resolution so that a pixel in a slice represents the same spatial in-formation in another slice and in the other modality. For this purpose and since the DE MRI is considered as the target modality, all the data was resized to correspond to the median spatial resolution of the DE images (1.87 mm) (in 2D). Zero-padding was then applied for the smallest volume of each couple of data (DE and cor-responding CINE), to have the same image size.

3.3.2. Contrast enhancement

MR images tend to need contrast enhancement for a better visual quality. In our case, Contrast-Limited Adaptive Histogram equalization (CLAHE) was ap-plied. This method is a variant of the ordinary adaptive histogram equalization for contrast improvement that prevents noise amplification. In CLAHE method, the contrast enhancement is local and based on regions with a predetermined size (neighborhood). Additionally, the histogram is clipped at a predefined value named "clip limit", this will limit the slop of the transformation func-tion, consequently it prevents the over-amplification of the contrast. We applied CLAHE on 2D DE images with a clip limit of 2 and a neighborhood of 8 x 8.

3.3.3. Image registration

Image registration can be defined as the process of aligning two or more images using a specific geometrical transformation. The aim of an image registration algorithm is to find the optimal transformation that aligns the structures of interest of the moving image to those of the fixed image. The main registration categories include rigid transformations that use simple operations such as translations, rotations and scaling, as well as more complex transformations with higher degrees of freedom such as affine transformation or deformable spline models. As previously mentioned, the CINEDE dataset contains two modalities with considerable shifts in the localization of the structure of interest (left ventricle of the heart). To address this problem, image registration is applied where the moving image is the CINE modality and the fixed image is the DE modality. Due to the complexity of the task, high degree transformations were omitted as the resulting deformations were too consequent and also because no deformation should be needed after the spacing is harmonized. Therefore, only a simple translation transformation was applied in the final registration pipeline to help better align the LV structures on the two modalities. Describing the transformation between a CINE image and a DE image by a translation transformation corresponds to the clinical perspective. Indeed, during the exam, CINE and DE are not acquired at the same moment (between two apneas). Even though the positioning should be the same, a translation in the images is expected due to a difference in acquisition time (i.e. different apnea). Moreover, the different acquisition time also results in slides not totally corresponding to the same heart regions for both modalities. The registration process aims to correct these problems.

The use of segmentation is very common in a medical image registration processes as it helps guide the registration on aligning the structures of interest. However, since segmentation is the end goal of our pipeline and we can not assume to have it at this early step, a region of interest (ROI) detection is applied in replacement of segmentation. The ROI detection aims to help focus the registration on the left ventricle region and guide the transformation towards LV correspondence in both modalities. Details of the ROI detection are explained next.

- ROI detection using Mask-RCNN :

  Mask-R CNN (He et al., 2017) is a convolutional neural network whose aim is to perform instance segmentation. This model outputs a class, a surrounding bounding box and a segmentation mask for each object in a given image. As in Faster-R CNN (Ren et al., 2015), it adopts two stages, the first stage is called a region proposal network (RPN) that proposes candidate object bounding boxes. The second stage consists of extracting fea-

tures from each candidate box to output the class, the bounding box offset and the mask (see Figure 3).
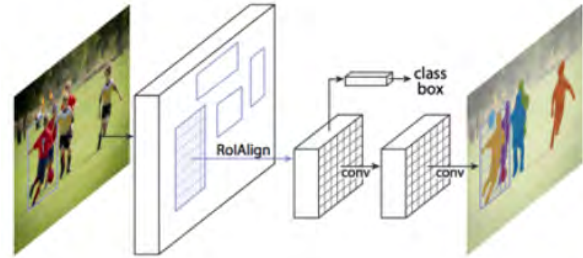
Figure 3: Mask R-CNN architecture

To detect the left ventricle on both CINE and DE MR images, a Mask-R CNN model from the torchvision library pretrained on the COCO dataset was fine-tuned using the training sets of both modalities, separately. Some customizations on the dataset were made to be able to run the training: the classes were narrowed down to 2 classes (object or no object), and bounding boxes were taken from the GT masks (including both cavity and myocardium as one object).

After the training phase, only the bounding boxes' coordinates outputs are considered. The LV region is then defined by applying a 10% margin on the bounding box with the highest prediction score since Mask-R CNN could detect more than one bounding box in the image.

- Application of the registration : After the ROI detection, A pixel based matching registration is applied which consists in finding the best transformation matrix (translation along x and y in our case, in order to avoid excessive deformations) that aligns the ROI of CINE to the ROI of DE. The interpolator used in the procedure is linear, and both Mattes mutual information and mean squares error metrics were tried, the better performance being reached using the latter one. For the optimization, a regular gradient descent optimizer was employed. Results and impact of the registration will be further discussed.

3.4. Fusion strategies

Multiple fusion strategies were investigated using 2D U-Net based architectures. The number of encoder-decoder blocks and number of filters were fixed for all the different architectures. There are 5 encoder blocks starting with 64 filters and ending with 1024 channels in the latent space. The input shape was also fixed to 256 by 256 grayscale images.

All architectures are defined by two outputs at the decision level, one output for the CINE segmentation mask and the other one is for the DE segmentation mask.

14.6

3.4.1. Input Fusion

Input fusion or FIUNet, is the simplest fusion approach. The architecture (Figure 4) consists of a simple concatenation of CINE and DE images at the input level. The remaining part of the network is a classic 2D U-Net architecture.

3.4.2. Output Fusion

In the output fusion architecture, unlike the first approach, two separate 2D U-Nets are trained each on one modality, and it is only at the decision level, before the last convolution layer that the feature maps from both modalities are concatenated, as it is illustrated in Figure 5. We will also designate this architecture by FOUNet in the next parts.

3.4.3. LFUNet

Layers fusion U-Net or LFUNet was inspired by FuseNet (Caner et al., 2016). The network consists of two identical encoders, one for the CINE modality and the other for the DE modality. The feature maps resulting from these two encoders are concatenated at each stage, where we concatenate at each stage the feature maps, meaning that at each resolution level the resulting feature maps have the shape of 2N x X x Y, where N is the original number of filters at each convolution level, and (X,Y) are the size depending on the encoder layer. In order not to change the decoder architecture, the number of convolutional filters after each concatenation remains the same, this way the expected number of feature maps is preserved. Figure 6 gives the general idea of this architecture.

3.4.4. Intermediate Fusion - DualUNet

This network was inspired from the work done in (ref), where T1 and T1 flipped MR images were used as the two inputs in a encoder-decoder architecture for brain tumor segmentation. In our case the inputs are replaced by our two cardiac MRI modalities. We can consider the fusion approach here as an intermediate stage fusion as it comes later on than the LFUNet (Figure 7). The fusion block (Figure 8) outputs fusion features that are going to be given at the start of the decoder path and at the skip connections levels. It works as follows: at each encoder level, we take the two outputted feature maps of size N x X x Y, where N is the number of convolutional filters and (X,Y) the size depending upon the encoder stage, and we stack them adding an extra dimension to have a shape of N x 2 x X x Y. The fusion is then made using a 2 x 1 x 1 3D convolution that will produce an output shape of N x 1 x X x Y. This output is squeezed to recover the original feature maps shape. The decoder part of the network is kept identical to the original U-Net.

3.4.5. Self Supervised module adaptation fusion - SS-MAUNet

In SSMAUNet, the fusion is based on the self supervised model adaptation fusion scheme mentioned previously in the multi-modal scene segmentation section (Valada et al., 2019). The architecture of this model is composed of two identical U-Net encoders and one decoder. The fusion is made at the skip connections level and at the input of the decoder by using the SSMA module (Figure 9). The SSMA module aims to model the correlation between the two modality specific feature maps, it is a convolutional path made of 2 convolutions of the stacked maps, the first one followed by a ReLu activation function and the second by a sigmoid activation to scale the dynamic range of activations between 0 and 1, the resulting output of this path is then multiplied by the original stacked feature maps which will enable the network to weight the features element-wise according to the spatial information and the channel depth (Figure 10), similar to classic spatial and channel attention.

3.5. Training details

3.5.1. Dataset split

For the experiments, the dataset was split into 3 sets :

- The training set : composed of 70% of the dataset, i.e. 84 patients which results in 667 slices for each modality.

- The validation set : It is used to monitor the training, composed of 20% of the dataset, i.e. 26 patients that results in 202 slices.

- The test set : It is used after the training phase to test the generalization ability of the model. It is composed of the remaining 10% of the dataset, ie. 14 patients that give 114 slices.

3.5.2. Optimization

Training deep neural networks requires tuning the hyper-parameters. Several trials were run to decide on an ensemble of fixed values of hyper-parameters used on all trials for the different architectures.

- Loss function: The loss function is responsible for quantifying the difference between the expected and the predicted outcome, and it is used by the optimizer to gradually update the network parameters through the training process. It is used to evaluate the performance on the validation set as well. For the segmentation task, multiple losses and their combinations were tried out such as cross entropy loss, dice loss, focal loss. At the end, the Monai DiceFocalLoss was used as it obtained the best results.

  The Dice Focal loss computes the Dice Loss and the Focal Loss and returns a weighted sum of the
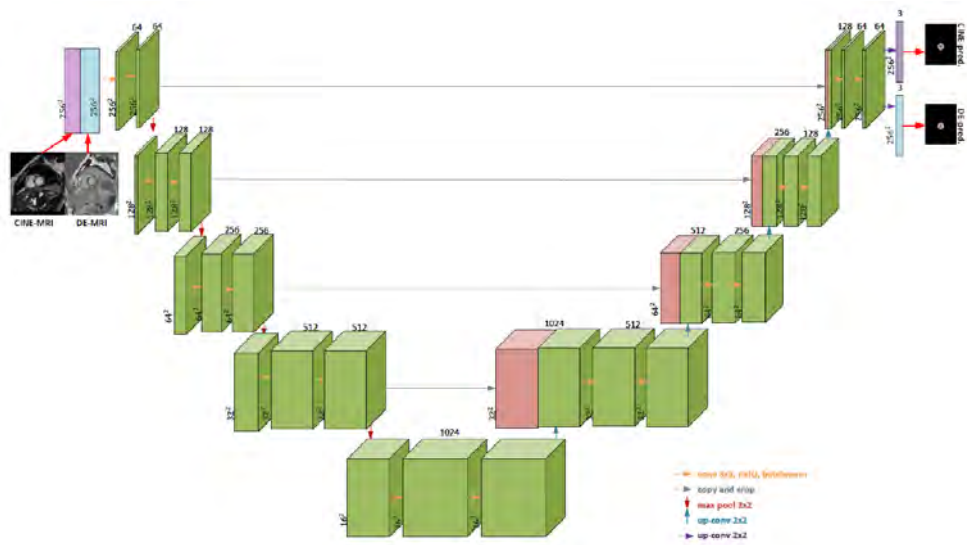
Figure 4: FIUNet architecture, where the two modalities are concatenated at the very input of the model (Hadadi, 2021)
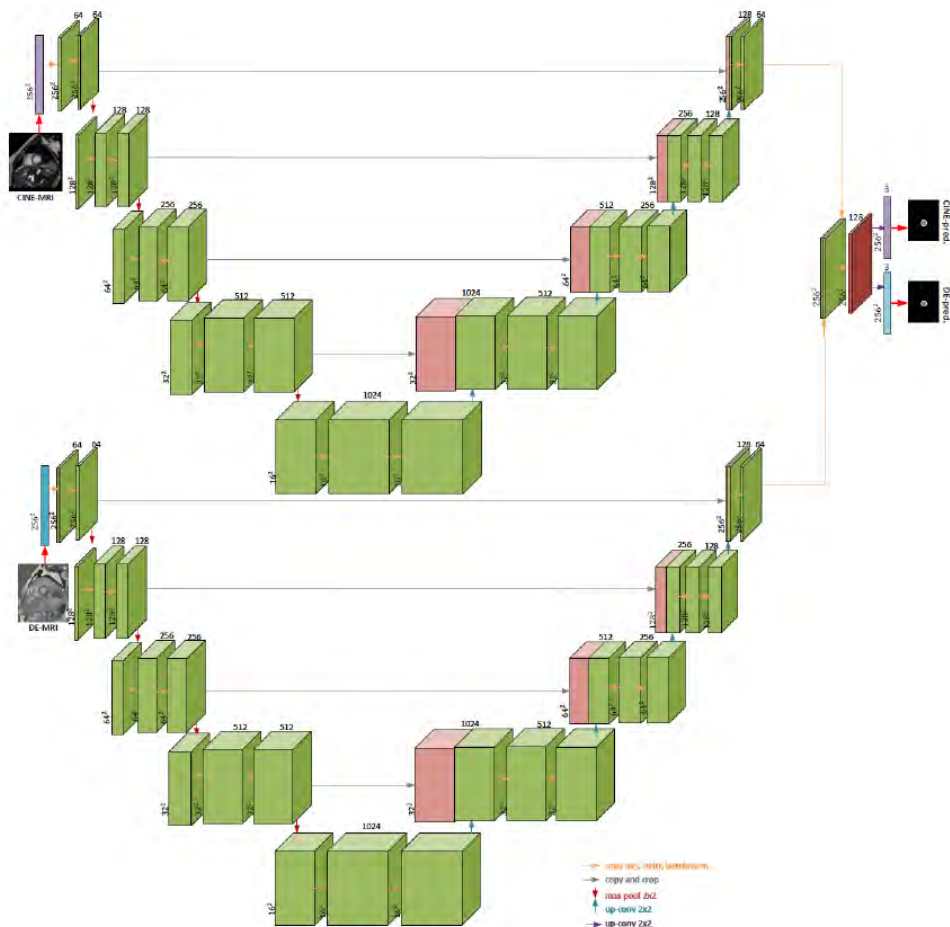
Figure 5: FOUNet architecture, where the two modalities are concatenated before the last convolution (Hadadi, 2021)
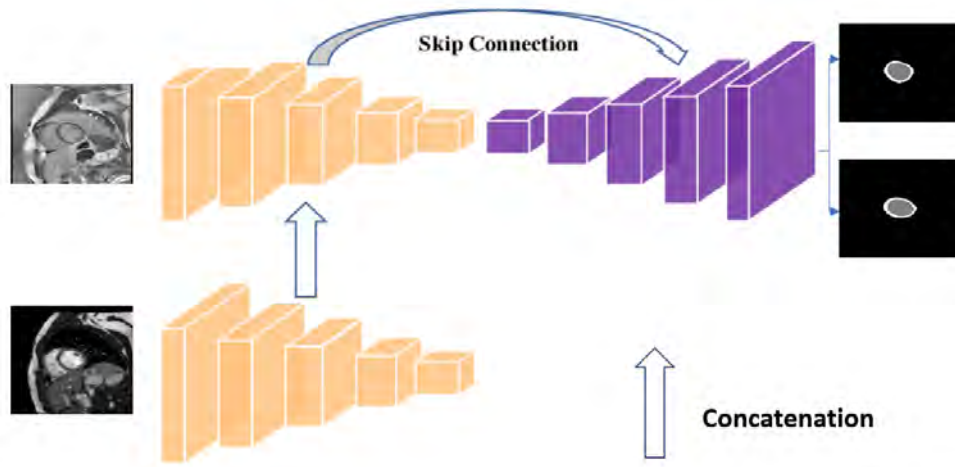
Figure 6: LFUNet architecture: feature maps are concatenated through all the resolution levels from one encoder to the other
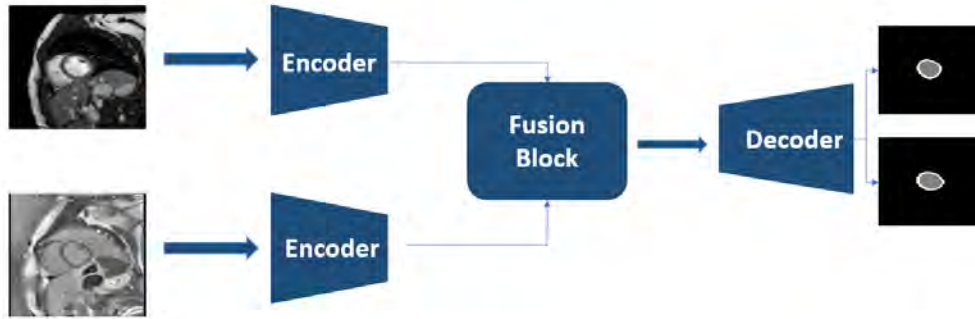
Figure 7: DualUNet architecture: The feature maps from the two encoders are fused through a fusion block before the decoder step

Figure 8: Fusion Block for resolution level i
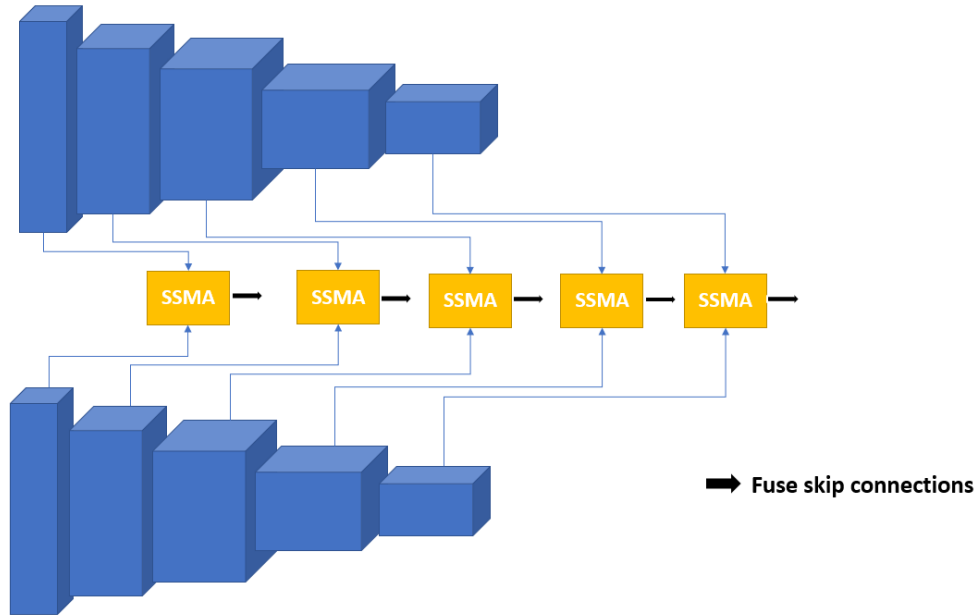
Figure 9: Modality fusion in SSMAUNet : The feature maps from the two encoders are fused through the SSMA block before the decoder step
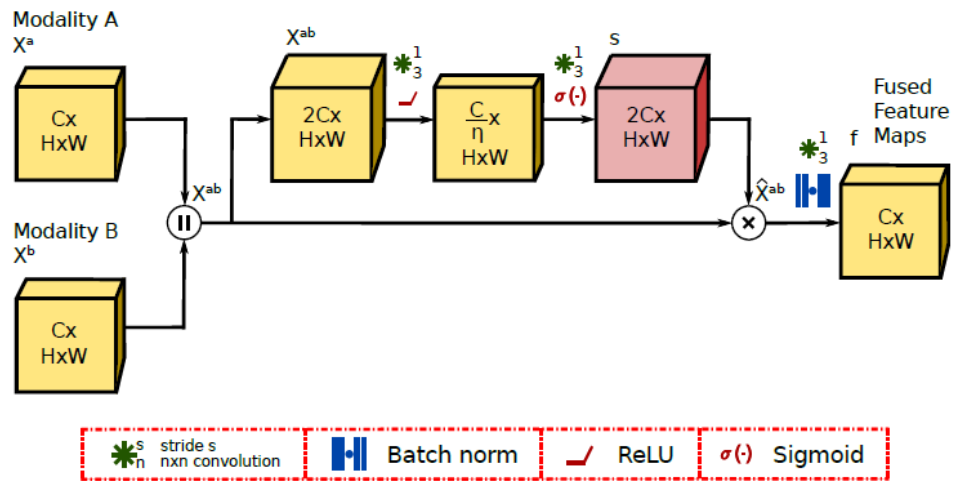
Figure 10: SSMA unit topology that adaptively fuse modality-specific feature maps based on the inputs. η denotes the bottleneck compression rate (Valada et al., 2019).

two losses. In our experiments, the same weight is given.

- Optimizer: The adaptive Moment Estimation (ADAM) (Diederik et al., 2015) optimizer was used with a learning rate of 10^{-4} and a weight decay of 10^{-5}.

- Batch size: Both 16 and 8 batch sizes were tried, and although the difference was not really significant, a batch size of 8 gave overall better results. Higher batch sizes could not fit the GPU memory for the fusion models.

- Early stopping: The models were trained without a limitation of epochs, but with an early stopping method monitoring the training over the validation loss with a patience of 10 epochs.

3.5.3. Data augmentation

Basic data augmentation was applied while training using the PyTorch DataLoaders. The transformations used were reduced to random orientations and random flips of the images and their corresponding masks. Random cropping and contrast changes were tried but did not provide better results.

3.6. Evaluation Metrics

3.6.1. Intersection over Union

The intersection over union (IOU) illustrated in Figure 11, is an index ranging between 0 and 1 computed by dividing the intersection area by the union of areas of two bounding boxes.
This metric was used to evaluate the ROI detection performance.
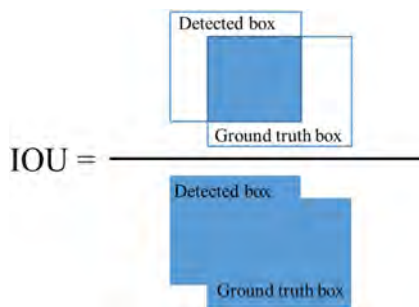
Figure 11: IOU

3.6.2. Dice Score

The Dice score (DSC) is the most frequently used metric for semantic segmentation evaluation. Considering GT as the ground truth area and S as the predicted area, the DSC is calculated as follow :

$$DSC = \frac{2|GT| \cap |S|}{|GT| + |S|}$$

Dice score was used to evaluate the segmentation performance for the two classes (cavity and myocardium), but it also provided information about the registration performance by calculating the DSC between the DE mask and the new CINE mask.

4. Results

This section is divided in three parts. The first part showcases the results of the LV ROI detection while the second part is dedicated to the registration results. The third part of the section is the main part of the work, it will focus on showing the segmentation results of the left ventricle on both CINE and DE modalities according to the fusion strategies.

4.1. ROI detection results

Table 1 displays the IOU measurements on the test set for both modalities (DE and CINE), obtained using Mask-RCNN to localize the heart region (specifically, the LV) :

	IOU
CINE	0.921
DE	0.916

Table 1: Intersection over union results on the test set

Some of the visual results of the bounding boxes for both modalities are shown in Figure 12 for CINE modality and Figure 13 for DE modality. The blue corresponds to the predicted bounding box by the Mask R-CNN, as for the green color, it corresponds to the ground truth bounding box taken from the GT masks.

4.2. Registration results

The DSC was used for the registration evaluation. The Dice score is calculated between the CINE mask and the DE mask for both classes (Left ventricle cavity and the myocardium). Table 2 displays the results of DSC before and after the registration. A definite improvement is observed after the translation.

	Before	After
CV	0.52	0.84
MYO	0.24	0.65

Table 2: DSC score before and after registration - CV : left ventricle cavity, MYO : myocardium.

Figure 14 shows the overlaying of the CINE masks on the DE images before and after the registration on two cases, one per line. We can see that after the registration process, the CINE mask became corresponding to the DE image, independently of the initial displacement.
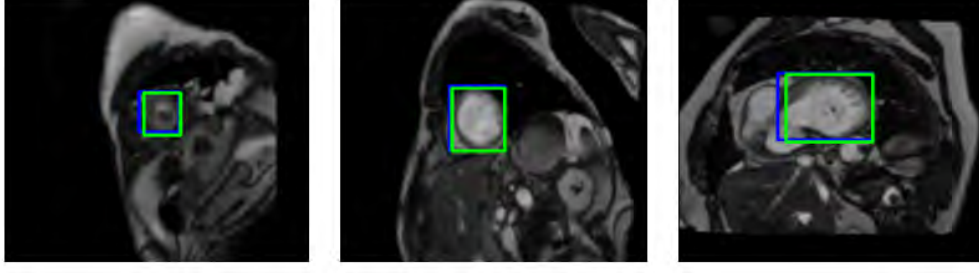
Figure 12: Mask R-CNN ROI detection on CINE modality(Apex slice- Middle slice - Basal slice), Green color refers to the GT and Blue to the predicted bounding box
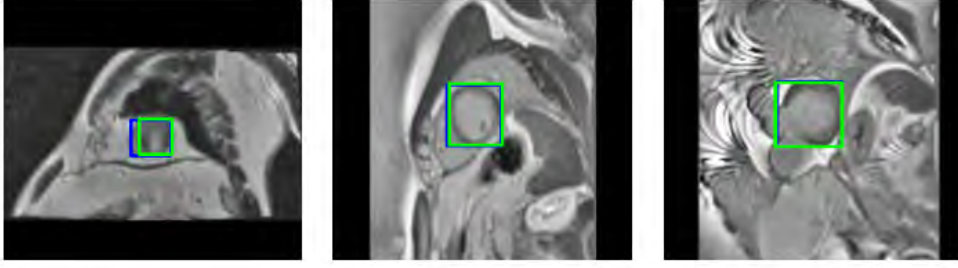
Figure 13: Mask R-CNN ROI detection on DE modality(Apex slice - Middle slice - Basal slice), Green color refers to the GT and Blue to the predicted bounding box
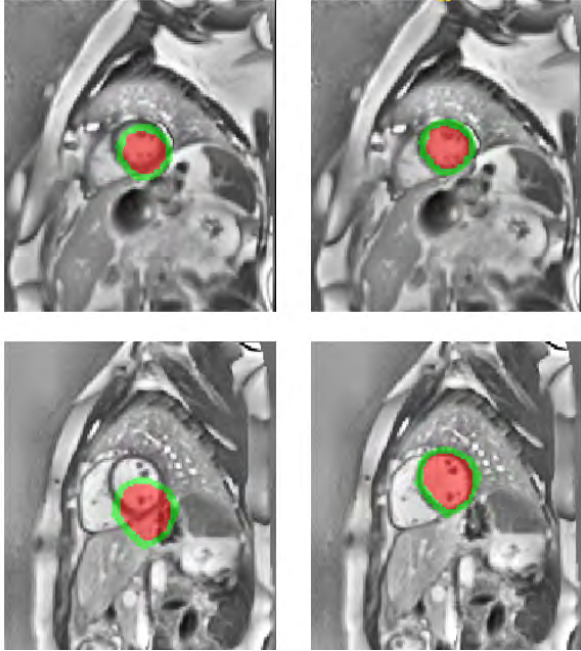
Figure 14: CINE mask applied on DE images after and before registration for two different initial displacements - Red: left ventricle cavity, Green: myocardium.

4.3. Segmentation results

This section will showcase the results of the different experiments, starting from the segmentation results using single modality (CINE and DE) to the different fusion strategies with and without registration. Finally,

the last experiment studies the impact of an ideal ROI detection.

We will emphasize on the DE segmentation results over the CINE modality as it is the main modality of interest in our work.

4.3.1. Single modality

The first two experiments were consists in using the conventional U-Net for the segmentation of both DE and CINE modality separately. Mean Dice score and standard deviation results are shown in Table 3.

These results will be used for comparison purposes in the next section.

The acronym "CV" refers to the left ventricle cavity, and "MYO" to the myocardium.

		Validation	Test
CINE	CV	0.95 ± 0.02	0.95 ± 0.02
	MYO	0.85 ± 0.03	0.83 ± 0.04
DE	CV	0.94 ± 0.02	0.90 ± 0.05
	MYO	0.84 ± 0.04	0.77 ± 0.10

Table 3: Mean Dice score and standard deviation (std) for single modality segmentation - CV : left ventricle cavity, MYO : myocardium.

From Table 3, we can observe that the results on the validation set are close for CINE and DE MRI. However, the drop of performance is considerably higher in the test set for DE MRI.

4.3.2. multi-modal segmentation

Similar training experiments were conducted on the non registered dataset using the five different fusion models.Results of the mean 3D Dice score and its corresponding standard deviation can be found in Table 4 for the validation set, and Table 5 for the test set.

The same experiments were made using the registered dataset. Results of mean DSC and its standard deviation for the five different fusion approaches are shown in Table 6 and Table 7, for validation and test sets respectively.

For the first group of experiments exploiting non registered data, FOUNet obtained the lowest Dice scores among all the fusion methods.
On the test set, DE Dice score is 0.63 ± 0.26 for the cavity and 0.57 ± 0.10 for the myocardium. The best performance was achieved by DualUNet, an intermediate fusion strategy, reaching a DE Dice score of 0.91 ± 0.05 for the cavity and 0.78 ± 0.10 for the myocardium. DualUNet is followed by SSMAUNet and LFUNet that were very close to the best model in terms of qualitative results.

From Table 6 and Table 7, we can observe the impact of the registration on the segmentation results. The performance of FOUNet encountered a considerable jump after registration. The DE Dice score went up to 0.90 ± 0.05 for the cavity, 0.71 ± 0.06 for the myocardium. Correspondingly with the first group of experiments, DualUNet obtained again the best qualitative results considering both mean Dice score and standard deviation with 0.93 ± 0.03 for the cavity, and 0.81 ± 0.06 for the myocardium. On every model, registration helped to increase the accuracy and robustness of the segmentation, as is best observed on the test set scores.

Even though the results of the models LFUNet, DualUNet and SSMAUNet are very close, the selected model used for additional qualitative results display, and comparisons is DualUNet as it holds the highest performance overall, and for both modalities.
Figures 15, 16 and 17 show segmentation masks produced by DualUNet overlaid on the DE images and with the corresponding ground truths. The figures display different examples from low performance to high performance. In addition, different regions of the heart are selected : apex slices (Figure 15) , middle slices (Figure 16) and basal slice (Figure 17).

To further investigate the potential advantage of multi-modal data on the myocardium segmentation on delayed enhancement MRI, we selected cases with less than 0.70 of DSC to isolate the worst cases on the single modality.

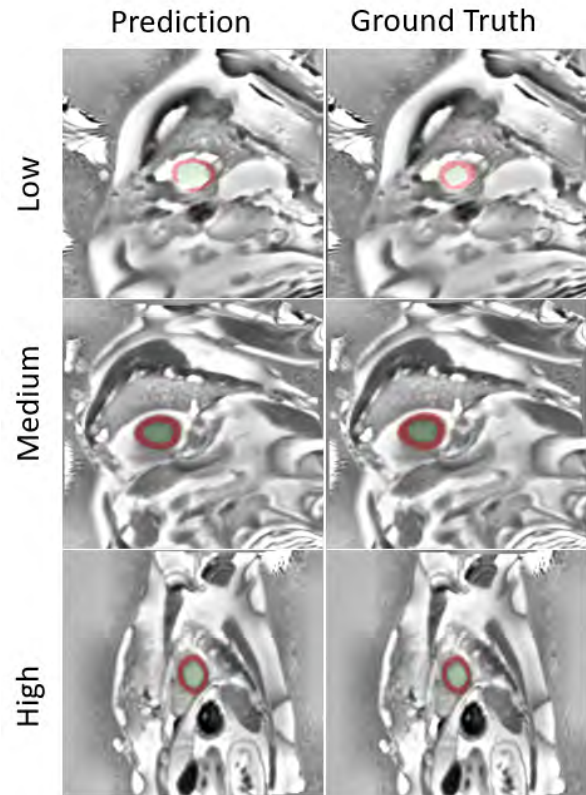Figure 15: DualUNet results on DE apex slices according to performance, Green: left ventricle cavity, Red: myocardium.
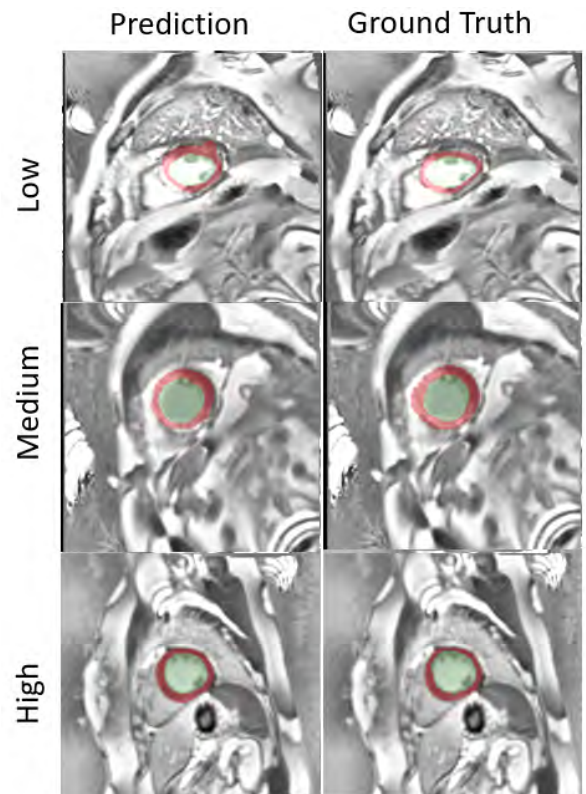

Figure 16: DualUNet results on DE middle slices according to performance, Green: left ventricle cavity, Red: myocardium.

		FOUNet	FIUNet	LFUNet	DualUNet	SSMAUNet
CINE	CV	0.69 ± 0.20	0.93 ± 0.03	0.95 ± 0.02	0.95 ± 0.01	0.95 ± 0.01
	MYO	0.66 ± 0.06	0.80 ± 0.05	0.85 ± 0.03	0.85 ± 0.03	0.85 ± 0.03
DE	CV	0.70 ± 0.25	0.93 ± 0.03	0.94 ± 0.02	0.95 ± 0.02	0.95 ± 0.02
	MYO	0.64 ± 0.09	0.79 ± 0.05	0.84 ± 0.04	0.85 ± 0.03	0.85 ± 0.04

Table 4: Validation set mean DSC and std results for multi-modal segmentation without registration - CV : left ventricle cavity, MYO : myocardium.

		FOUNet	FIUNet	LFUNet	DualUNet	SSMAUNet
CINE	CV	0.74 ± 0.15	0.93 ± 0.02	0.94 ± 0.02	0.95 ± 0.01	0.95 ± 0.01
	MYO	0.64± 0.06	0.77 ± 0.07	0.82 ± 0.05	0.83 ± 0.04	0.82 ± 0.04
DE	CV	0.63 ± 0.26	0.88 ± 0.06	0.90 ± 0.06	0.91 ± 0.05	0.91 ± 0.05
	MYO	0.57 ± 0.10	0.69 ± 0.12	0.75 ± 0.12	0.78 ± 0.10	0.77 ± 0.12

Table 5: Test set mean DSC and std results for multi-modal segmentation without registration - CV : left ventricle cavity, MYO : myocardium.

		FOUNet	FIUNet	LFUNet	DualUNet	SSMAUNet
CINE	CV	0.90 ± 0.03	0.94 ± 0.02	0.95 ± 0.01	0.95 ± 0.01	0.95 ± 0.01
	MYO	0.79 ± 0.05	0.81 ± 0.04	0.84 ± 0.03	0.85 ± 0.03	0.85 ± 0.03
DE	CV	0.92 ± 0.03	0.92 ± 0.03	0.94 ± 0.02	0.95 ± 0.02	0.95 ± 0.02
	MYO	0.78 ± 0.06	0.80 ± 0.07	0.85 ± 0.04	0.86 ± 0.04	0.86 ± 0.04

Table 6: Validation set mean DSC and std results for multi-modal segmentation with registration - CV : left ventricle cavity, MYO : myocardium.

		FOUNet	FIUNet	LFUNet	DualUNet	SSMAUNet
CINE	CV	0.91 ± 0.03	0.94 ± 0.03	0.94 ± 0.03	0.95 ± 0.01	0.95 ± 0.01
	MYO	0.78 ± 0.04	0.79 ± 0.06	0.82 ± 0.04	0.83 ± 0.03	0.83 ± 0.04
DE	CV	0.90 ± 0.05	0.90 ± 0.05	0.92 ± 0.04	0.93 ± 0.03	0.93 ± 0.04
	MYO	0.71 ± 0.06	0.75 ± 0.12	0.80 ± 0.06	0.81 ± 0.06	0.80 ± 0.09

Table 7: Test set mean DSC and std results for multi-modal segmentation with registration - CV : left ventricle cavity, MYO : myocardium.

Table 8 shows the Dice score performance on the 2 worst cases, using single modality (U-Net) and DualUNet (as it is the selected model). As can be seen, with the DualUNet segmentation, the Dice score of the myocardium went from 0.63 to 0.72 for case 1, and from 0.51 to 0.70 for case 2.

		Case 1	Case 2
Single	CV	0.88	0.84
	MYO	0.63	0.51
multi-modal	CV	0.92	0.91
	MYO	0.72	0.70

Table 8: Dice score for the two worst cases using U-Net and DualUNet - CV : left ventricle cavity, MYO : myocardium.

Representative visual results of these two hard cases, for both single modality and DualUNet, can be seen in Figure 18 for case 1, and in Figure 19 for case 2.

Figure 20 illustrates the box-plots for the myocardium Dice scores on the test sets, on all the five fusion strategies in addition to the single modality model (U-Net). From the distributions, we observe that the intermediate fusion models are more robust, particularly DualUNet where the mean is the highest, and the variation is the lowest among all the models. FOUNet has the worst results in terms of Dice values, however the length of the box-plot is shorter than the one of single modality, which is a marker of more robustness.

As stated before, the CINEDE dataset contains cases of different pathologies. Even though not all of them are included in the test set, we summarized in Table 9 the average Dice score obtained with single modality and DualUNet according to the existing pathologies in the test set. We notice that the performance on the normal and myocarditis cases is quite similar for both models. However, with DualUNet, there is a consider-
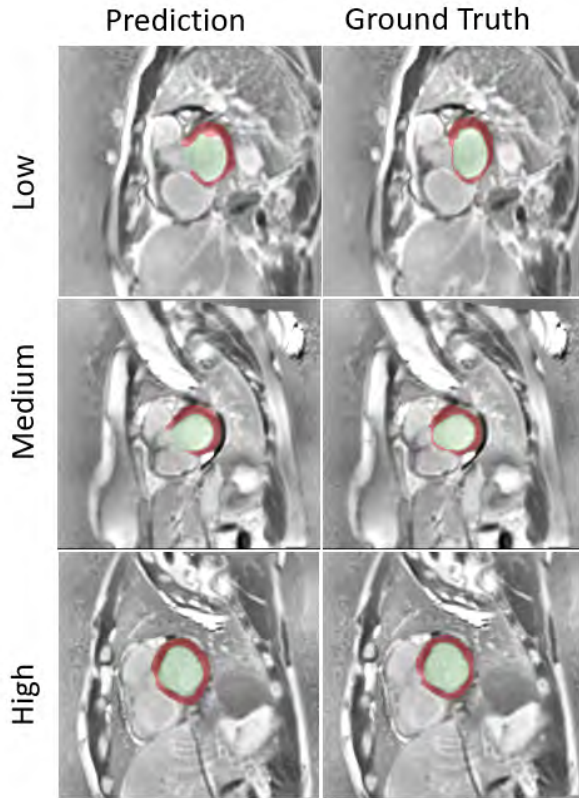
Figure 17: DualUNet results on DE basal slices according to performance, Green: left ventricle cavity, Red: myocardium.
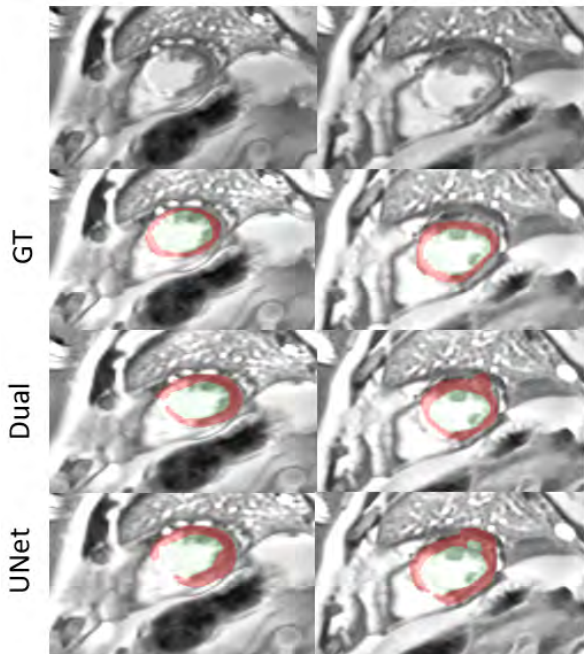
Figure 18: Single and multi-modal segmentations for Case 1, Green: left ventricle cavity, Red: myocardium.

able increase in the cases of MI, particularly in the myocardium Dice score that goes from an average of 0.72 to 0.78. We can also observe a slight improvement of
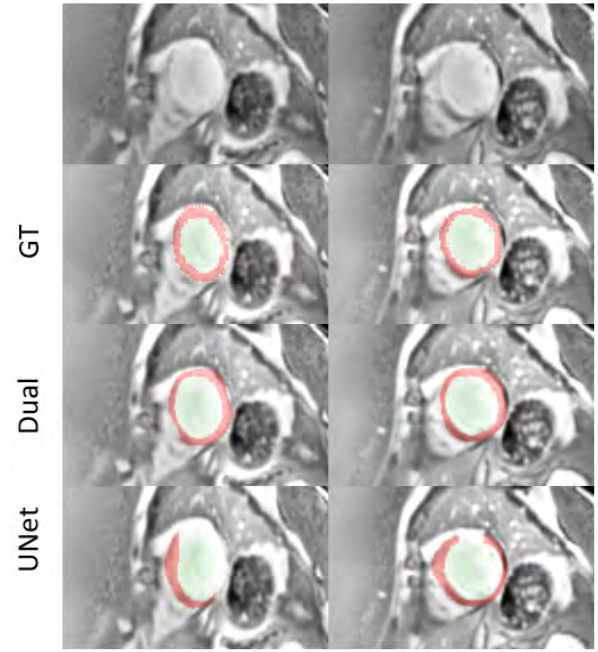
Figure 19: Single and multi-modal segmentations for Case 2, Green: left ventricle cavity, Red: myocardium.

the cavity segmentation in case of dilated cardiomyopathy (CMD).

		Normal	MI	CMD	Myo-C
Single	CV	0.96	0.89	0.90	0.95
	MYO	0.87	0.72	0.80	0.86
DualUNet	CV	0.95	0.92	0.93	0.95
	MYO	0.86	0.78	0.80	0.87

Table 9: Average Dice score according to existing pathologies in the test set, with CMD refering to dialted cardiomyopathy, MI to myocardial infarction and Myo-C to myocarditis - CV to left ventricle cavity and MYO to myocardium

Segmentation after ROI detection :

The last experiment was made to explore the potential performance improvement when limiting the area of the interest to the left ventricle region. We selected DualUNet according to the previous results for this part. For ROI detection, two trials were run :

- Using ideal ROIs extracted from the groundtruth with a margin of 10% dedicated to fully preserve the edges.

- Using ROIs extracted from Mask-RCNN, with a 15% margin.

In both trials, resizing the ROI into 256 x 256 makes the image too pixelated. Thus, we resized the ROI to 128 x 128 and completed with zero padding to recover
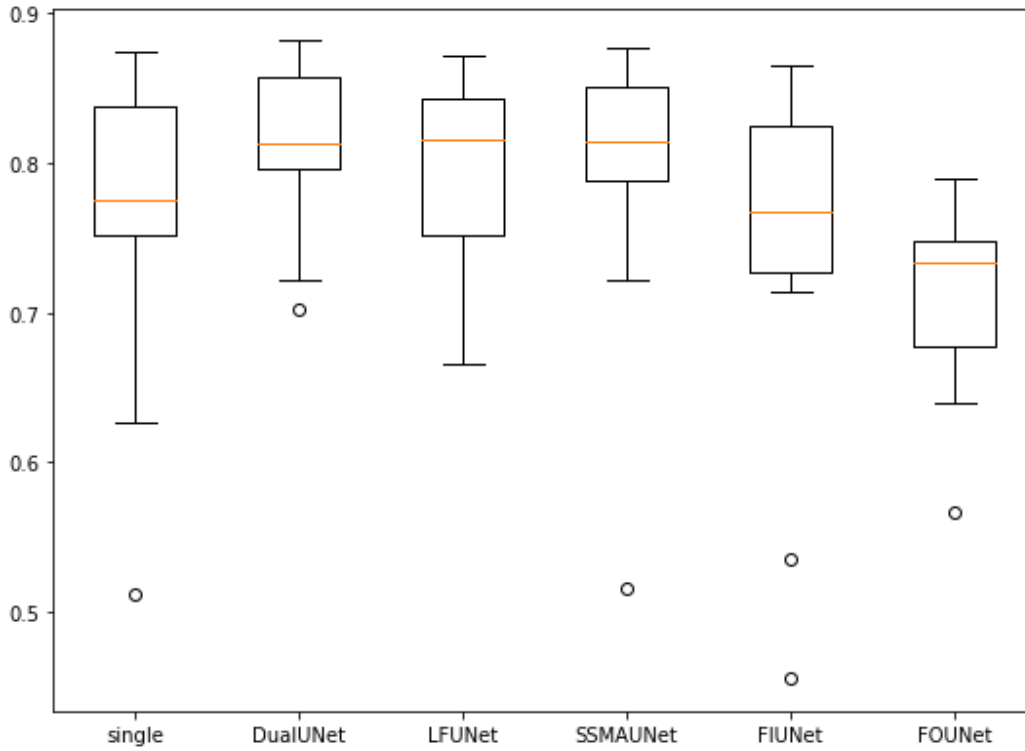
Figure 20: Box-plot illustrating the distribution of the myocardium Dice scores on the test set for all models

the model input size of 256 x 256.

Figures 21 and 22 show the input images after an ideal ROI detection and Mask R-CNN ROI detection respectively. These figures also demonstrate that when using Mask R-CNN the cropped heart region may vary between CINE and DE modalities for the same case. These variations occur also within the DE slices and the CINE slices of a same volume.

	Ideal ROI	Mrcnn ROI
CV	0,94 ± 0.04	0.91± 0.04
MYO	0.86 ± 0.04	0.78± 0.07

Table 10: UNet test set Dice score results using ideal and Mask R-CNN ROI - CV : left ventricle cavity, MYO : myocardium.

		Ideal ROI	Mrcnn ROI
CINE	CV	0.96 ± 0.01	0.94 ± 0.01
	MYO	0.86 ± 0.01	0.83 ± 0.03
DE	CV	0.94 ± 0.03	0.91± 0.04
	MYO	0.86 ± 0.04	0.79± 0.08

Table 11: DualUNet test set Dice score results using ideal and Mask R-CNN ROI - CV : left ventricle cavity, MYO : myocardium.

Table 10 shows the Dice scores obtained on the test set with the single modality (DE), when using both ideal ROI and Mask R-CNN detected ROI. Additionally, Ta-
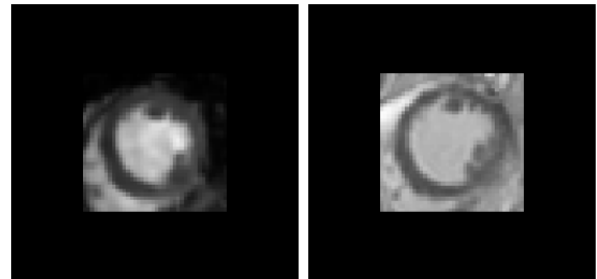
Figure 21: Example of DE and CINE slices with ideal ROI detection
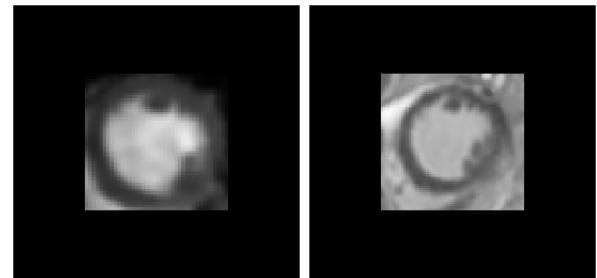
Figure 22: Example of DE and CINE slices with Mask R-CNN ROI detection

ble 11 summarizes the test set results for both CINE and DE images. These latter results show an improvement of performance when using an ideal ROI, but this does not apply to the results using Mask R-CNN ROI detection. Moreover, the DualUNet seems to have very similar results with the single modality approach , im-

14.16

plying that intermediate fusion has a similar effect on DE segmentation than a pre-localization of the heart.

5. Discussion

In this work, we investigated several fusion strategies of CINE and DE MRI for left ventricle (LV) segmentation. The first experiments aimed at providing reference results when segmenting both CINE and DE separately using the conventional U-Net. From Table 3, we can see that the overall performance was similar on DE and CINE for the validation set, however, for the test set, the Dice score on DE dropped significantly by 5% for the cavity and 7% for the myocardium compared to the validation. We also note that the standard deviation is significantly higher for the DE modality. These first results imply that a good generalization is harder to achieve on the DE modality than the CINE, using data provided from the same exams. This is may be due to a higher variability in the DE images because for example of the presence of myocardial infarction. This established generalization ability difference can be considered as a valid motive to use multi-modal fusion strategies to perform DE left ventricle segmentation.

Given the differences between the DE slices and their corresponding CINE slices, such as slices orientation but mainly different displacements of the heart region on the images. We conducted experiments on five different fusion strategies using the CINEDE dataset before and after registration. Table 4 and Table 5 summarized all the results of the five fusion strategies on the non registered CINEDE. Output fusion (FOUNet) gave the worst results among all methods, followed by the input fusion (FIUNet). Their efficiency however, improved using the registered dataset, especially when it comes to FOUNet (Table 6 and Table 7). This can be explained by the fact that in output fusion, there are two independent segmentation paths, as the fusion of the two modalities is made at the last step through a single and last convolution operation. Consequently, if the positions of the heart do not correspond on both modality, the network will tend to produce confusing segmentation masks. As the variation of the heart placements from both modalities is not controlled, the underlying relation between the two modalities would be harder to exploit with the output fusion scheme. In addition, we attest that the registration impact is notable for all other methods but less significant than for the output fusion. Overall , we can conclude that applying registration on the Dataset before training leads to producing better segmentation results, and more particularly, to a better generalization.

Comparing the different strategies, we find that the performance of the output fusion is very close but less good than the one of the input fusion. This finding contradicts the study (Zhou et al., 2019) that implies that

fusion at the output would more frequently be a better strategy than the fusion at the input, considering that in this case we have separate CNN expert on each modality, and that the second modality comes as additional external information. This contradiction may come from the misalignment of the information between the two images.

Additionally, both simple fusion strategies (i.e. input and output fusions) lead to worse performance than single modality. However, this does not imply that the information of the CINE in this case is not relevant for the segmentation task of DE but rather that it is not straightforward. Based on the higher results of ideal ROI segmentation (Table11), where a nearly perfect LV localization is used, and on the results summarized in Table7, improving the registration can be considered in order to improve the results for fusion models, including simple fusion strategies. However, a non rigid or a deformable registration would not be a wise choice from a clinical point of view. Hence, the main lead for such improvement come to improving the localization and adding data augmentation to reduce outliers.

According to both Table 6 and Table 7, the intermediate fusion models DualUNet and SSMAUNet and the fusion at the encoder level LFUNet outperformed the simple fusion strategies. This is in agreement with the work previously done (Zhou et al., 2019) and (Y.Zhang et al., 2021) which demonstrated that a layer level fusion is more effective and robust than input and output fusions. Furthermore, these models often gave better results than the single modality. For the best selected model DualUNet, the mean Dice score is better than the single modality for both validation and test sets. On top of that, the standard deviation of the myocardium DSC is lower for DualUNet. From the latter results, we can point out that the addition of the CINE MRI may bring more generalization ability and stability to the segmentation task. However, further experiments should be run to conclude on this.

In general, whether it is the single modality or multi-modality strategy, the hardest cases of DE myocardium segmentation are the ones including a myocardial infarction. Because of their strong clinical interest, these cases are especially taken into consideration. According to the results in Table 8, the best multi-modal strategy (DualUNet) seems to improve the performance on the worst cases (we note that both cases are MI cases). We can observe from the box-plot figure that overall, the performance of the myocardium segmentation is more robust in DualUNet than in the conventional U-Net. These latter results are promising and they indicate that by adding the CINE information, the myocardial border tends to be better detected, particularly in case of MI. This would coincide with the hypothesis made as the research motive of this study

From Table 9, we notice an improvement of myocardium segmentation in case of MI with DualUNet. Thus, we can conclude that multi-modality tends to help the myocardium segmentation in presence of MI. Additionally, according to the same table, multi-modality potentially helps the left ventricle cavity segmentation in case of dilated cardiomyopathy, which is a pathology that changes the heart shape.

In this work, the split of the data was made according to available information on pathologies but since we now had recovered the information for all the cases, a more balanced split of the data is intended for future experiments.

Finally, in the last experiment, results with ideal ROIs found in Table 10 and Table 11 show a robustness improvement. Indeed, the Dice score rises up to 0.86 for the myocardium on the test set. However, it has to be duly noted that these scores were calculated with respect to the cropped images, hence the average Dice is expected to drop going back to the original resolution. Furthermore, unlike the case with whole images as inputs, results from single and multi-modal strategies seem to be more or less similar in terms of average Dice and standard deviations. Additional trials need to be made to conclude solidly on this point. On the other hand, results using the Mask R-CNN detected ROIs did not improve the results, on the contrary the average Dice score dropped slightly from the whole images trial. As we can see in Figure 21 and Figure 22 , the ROIs detected on DE and CINE may not represent the exact same heart region (unlike ideal ROIs taken from the ground truth). This could be a significant factor in the drop of the segmentation performance compared to ideal ROIs segmentation. Finally, results using DualUNet on the whole images are not that far behind the results obtained on ideal ROIs. This would suggest that multi-modality could help to better localize the heart in the segmentation task, and replace the step of LV localization. Future work will focus more on this particular part. Optimizing the Mask R-CNN ROI detection and using different resizing methods using ROIs are potential changes.

In all the experiments made, the evaluation relied on the average Dice score and its standard deviation on top of visual results. Dice score is a reliable measure, especially when it comes to strategies comparisons, yet it is interesting to introduce a more local measure such as the Hausdorff distance (HD) to better evaluate the resulted automatic delineation.

6. Conclusions

This work was conducted in order to use two MRI modalities for DE segmentation improvement , therefore the evaluation was made on a private new dataset. The clinical use of the CINE segmentation led the research to focus on a multi-task segmentation. DualUNet

gave promising results in both DE and CINE segmentation, as the metrics of the validation set are comparable to those found in the literature. Moreover, DualUNet seems to improve the segmentation of the myocardium particularly in MI cases. For the myocardium segmentation on DE, DualUNet achieved 0.86 mean DSC when the state of the art from the EMIDEC challenge (using a dataset of 150 patients) is at 0.879. However, this latter result dropped notably on the test set. This drop could be explained by the limited amount of data, and the heterogeneity of the CINEDE dataset, as the dataset consists of several pathological cases in addition to normal ones, in opposite to the EMIDEC dataset that includes only normal and MI exams.

Presence of MI in the DE images make the myocardium segmentation more challenging. In addition, there is a significant presence of noise and other artifacts due to the acquisition. In order to improve the results, data augmentation can be further investigated by introducing synthetic cases of MI produced from modifications of normal cases among the dataset. Overall performance improvement is a part that we want to look into as well, by making modifications on the base U-Net architecture. 3 D and 2.5 D models are also potential modifications that we would like to inspect in our future work.

References

A. Valada, G.L.O., 2017. Deep multispectral se1020 mantic scene understanding of forested environments using multimodal fusion. International Symposium on Experimental Robotics, Springer , pp. 465–477.

Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.A., Cetin, I., Lekadir, K., Camara, O., Gonzalez Ballester, M.A., Sanroma, G., Napel, S., Petersen, S., Tziritas, G., Grinias, E., Khened, M., Kollerathu, V.A., Krishnamurthi, G., Rohé, M.M., Pennec, X., Sermesant, M., Isensee, F., Jäger, P., Maier-Hein, K.H., Full, P.M., Wolf, I., Engelhardt, S., Baumgartner, C.F., Koch, L.M., Wolterink, J.M., Išgum, I., Jang, Y., Hong, Y., Patravali, J., Jain, S., Humbert, O., Jodoin, P.M., 2018. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? IEEE Transactions on medical imaging doi:37(11):2514-2525.

C. Petitjean, D.J., 2011. A review of segmentation methods in short axis cardiac mr images. medical imaging analysis , 169–184.

Caner, H., Lingni, M., Csaba, D., Daniel, C., 2016. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. Asian Conference on Computer Vision (ACCV) doi:10.1007/978-3-319-54181-5_14.

Cheng, Y., Cai, R., Li, Z., , Zhao, X., Huang, K., 2017. Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern 1110 Recognition , pp 3029–3037.

Couprie, C., Farabet, C., Najman, L., LeCun, Y., 2013. Indoor semantic segmentation using depth information. doi:10.48550/arXiv.1301.3572.

Deng, L., Yang, M., Li, T., He, Y., Wang, C., 2019. Rfbnet: Deep multimodal networks with residual fusion blocks for rgb-d semantic segmentation. 1090 arXiv preprint doi:arXiv:1907.00135.

Diederik, P., Kingma, Ba, J., 2015. Adam: A method for stochastic optimization. The 3rd International Conference for Learning Representations, San Diego doi:10.48550/arXiv.1412.6980.

Guo, Z., Li, X., Huang, H., Guo, N., Lia, Q., 2019. Deep learning-based image segmentation on multimodal medical imaging. IEEE Transactions on Radiation and Plasma Medical Sciences 3, pp. 162–169. doi:10.1109/TRPMS.2018.2890359.

Hadadi, A., 2021. Automatic segmentation of the myocardium from multi-modal mri with deep learning. Master Thesis .

Hazirbas, C., L. Ma, C.D., Cremers, D., 2017. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. 010 Lecture Notes in Computer Science (including subseries Lecture Notes in Articial Intelligence and Lecture Notes in Bioinformatics) 10111.

He, K., Gkioxari, G., Dollár, P., Girshick., R., 2017. Mask r-cnn. IEEE International Conference on Computer Vision (ICCV) , 95–107doi:10.1109/ICCV.2017.322.

Hu J., Shen L., S.G., 2018. Squeeze-and-excitation networks. IEEE/CVF Conference on Computer Vision and Pattern Recognition , 7132–7141doi:10.1109/CVPR.2018.00745.

K. Brahim, A.Q., Lalande, A., Boucher, A., Sakly, A., Meriaudeau., F., 2021. A 3d network based shape prior for automatic myocardial disease segmentation in delayed-enhancement mri irbm. Elsevier doi:10.1016/j.irbm.2021.02.005.

K. He, Z.X., S., R., J., S., 2016. Deep residual learning for image recognition. IEEE Conference on Computer Vision and Pattern Recognition, CVPR , 770–7781doi:10.1109/CVPR.2016.90.

KIM, R.J., S., F.D., PARRISH, T.B., HARRIS, K., CHEN, E.-L., S.O.B.J.F.J.P.K.F.J., JUDD, R.M., 1999. Relationship of mri delayed contrast enhancement to irreversible injury, infarct age, and contractile function. Circulation doi:10.1161/01.cir.100.19.1992.

Lalande, A., Chen, Z., Pommier, T., Decourselle, T., Qayyum, A., Salomon, M., Ginhac, D., Skandarani, Y., Boucher, A., Brahim, K., de Bruijne, M., Camarasa, R., Correia, T.M., Feng, X., Girum, K.B., Hennemuth, A., Huellebrand, M., Hussain, R., Ivantsits, M., Ma, J., Meyer, C., Sharma, R., Shi, J., Tsekos, N.V., Varela, M., Wang, X., Yang, S., Zhang, H., Zhang, Y., Zhou, Y., Zhuang, X., Couturier, R., Meriaudeau, F., 2022. Deep learning methods for automatic evaluation of delayed enhancement-mri. the results of the emidec challenge. Elsevier Medical imaging analysis doi:https://doi.org/10.1016/j.media.2022.102428.

Marchesseau, S., Ho, J.X., Totman, J.J., 2016. Influence of the short-axis cine acquisition protocol on the cardiac function evaluation: A reproducibility study. European Journal of Radiology Open 3, 60–66. doi:https://doi.org/10.1016/j.ejro.2016.03.003.

Oktay, O., Schlemper, J., Le Folgoc, L., L.M.H.M.M.K.M.K.M.S.Y.H.N.K.B.e.a., 2018. Attention u-net: learning where to look for the pancreas. arXiv e-prints .

R. Karim, B.P., P., C., R, J.H., Z, C., Z, K., HM, S., L, L.R., S, V., X, A., A, H., HO, P., T, A., MA, G.B., AF, F., M, G., R, R., T, S., K., R., 2016. Evaluation of state-of-the-art segmentation algorithms for left ventricle infarct from late gadolinium enhancement mr images. Med. Image Anal. 30, 95–107. doi:10.1016/j.media.2016.01.004.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. MIT Press .

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation .

Valada, A., Mohan, R., Burgard, W., 2019. Self-supervised model adaptation for multimodal semantic segmentation. International Journal of Computer Vision (IJCV), Special Issue: Deep Learning for Robotic Vision 128-5, 1239–1285. doi:10.48550/arXiv.1808.03833.

Y. Zhang, P.A.t.E., M., P., M., S., Campello, V., A., L., K., L., aputra A., S., O., C., Young, A., 2021. Cascaded convolutional neural network for automatic myocardial infarction segmentation from delayed-enhancement cardiac mri. Statistical Atlases and Computational Models of the Heart. MMs and EMIDEC Challenges , 328–333doi:10.1007/978-3-030-68107-4_33.

Y.Zhang, D.S., Morel, O., Mériaudeau, F., 2021. Deep multimodal fusion for semantic image segmentation: A survey. Elsevier doi:10.1016/j.imavis.2020.104042.

Zhou, T., Ruan, S., Canu, S., 2019. A review: Deep learning for medical image segmentation using multi-modality fusion. ELSEVIER doi:10.1016/j.array.2019.100004.

Deep learning pipeline for improved breast cancer detection in MRI

Santiago Pires, Supervisors: Koen Eppenhof, Mehmet Dalmis

ScreenPoint Medical - santiagopires13@gmail.com

Abstract

Even though dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) is the most sensitive modality for detecting breast cancer, there is no artificial intelligent system currently available in clinical practice that supports the radiologist by augmenting accuracy and increasing productivity. In this dissertation, we propose a three stage pipeline capable of detecting malignant lesions with high sensitivity at a low number of false positives. At the first step, fibroglandular and fatty tissue are segmented by 3D U-Net and UNETR architectures for performance comparison. The output volumes are then used to report the density values and mask the breast area in the second stage. In this step, a Retina-Unet is implemented to detect malignant lesion candidates on the relative enhancement volume. Finally, the classification stage utilizes information from other sequences in breast MRI to assign a probability of malignancy for each candidate, also working as a false positive reduction method. The use of a ResNet18 or Vision Transformer pre-trained in a Self Supervised Learning approach is discussed at this stage. Overall, the pipeline achieves a CPM score of 0.932(0.89-0.964), outperforming the previously proposed methods on a similar dataset. This exploratory work provides the basis for the development of an automatic CADe system that could be clinically deployed.

Keywords: DCE-MRI, Segmentation, Detection, Classification, Density, U-Net / UNETR, Retina-Unet, ResNet18, ViT, SSL, Occlusion sensitivity

1. Introduction

Cancer is a leading cause of death worldwide, accounting for nearly one in six deaths. According to the World Health Organization (WHO), breast cancer is the most common type of cancer in women with a total of 2,261,000 cases and 685,000 deaths worldwide in 2020 (Sung et al., 2021). Breast cancer occurs because of the abnormal growth of cells in the breast. The malignant tumors (cancerous) can be in-situ or invasive carcinoma. Ductal carcinoma in situ (DCIS) solely affects the mammary duct lobule system and remains confined to the layer of cells where it began. But the most harmful type of breast cancer, invasive carcinoma, can spread to other organs (Nassif et al., 2022). The benign tumors are a minor change in the breast structure (non-cancerous) that does not metastasize.

To avoid complications, it is vital to discover breast cancer early and correctly diagnose whether a tumor is benign or malignant. Medical imaging screening programs play a major role in reducing mortality and enabling easier treatment by detecting breast cancer before the symptomatic phase. Furthermore, imaging is also critical for breast cancer diagnosis, treatment and evaluation.

Digital mammography (DM) is the most popular breast imaging technique based on X-rays. It is a fast and easy technique, however, it suffers from the problem of tissue superposition. Especially in dense breasts with a high percentage of fibroglandular tissue (FGT), there are high chances that superposition of FGT hides or mimics lesions. As described in Mann et al. (2019b) for women with dense breasts, up to 50% of cancers are interval cancers, which means that they are detected in between screening rounds. This is significantly higher than the percentage in the general population.

High-risk patients are more likely to develop the disease earlier in life and are thus screened at a younger age when breast tends to be denser. Given that masses are more likely to be missed by mammography in this case, a more sensible imaging technique is needed for these patients.

Figure 1: Components of a non fat suppressed breast MRI protocol. In general, there is a non-contrast enhancement T2w and diffusion-weighted imaging, but in some cases for screening, this protocol may be abbreviated and one or both of them are not acquired. Regardless of the situation, a T1w and the contrast-enhanced series is acquired.

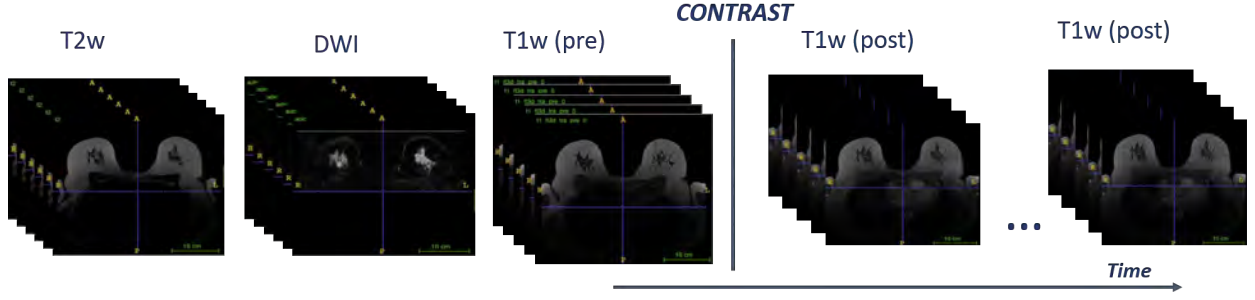Dynamic Contrast-Enhanced Magnetic Resonance Imaging (DCE-MRI) evaluates the permeability of blood vessels by using gadolinium as an intravenous contrast agent. The phenomenon behind contrast absorption by the cancer tissue is related to angiogenesis, a physiological process of the formation of new vessels. Since cancer tissue needs nutrients to grow quickly, it has a different vessel structure with highly permeable capillary walls and increased vascularity. When the tissue takes up the contrast agent, the T1 relaxation time shortens leading to a rapid local enhancement. As shown in Figure 1, during 5-7 minutes after the contrast administration, a series of T1 volumes are obtained (usually 3 to 5 post-contrast volumes) as the change of the signal intensity through time gives information about the type of lesion. A persistent increase is most commonly seen in benign lesions, whereas a decrease in the late phase is related to malignant lesions (washout). The common practice is to classify signal intensity curves in the late phase into 3 types: type 1 (persistent), type 2 (plateau) and type 3 (washout). While the basis relies on the T1-weighted sequence, breast MRI has evolved into a multiparametric technique, in which in some cases a T2-weighted and a diffusion volume (DWI) are performed. The contribution of T2-weighted is still debatable as several studies have reported that it improves specificity (Arponen et al., 2016) while other researchers have questioned if this modality adds any benefit in routine breast MRI(Mann et al., 2019a).

Although being a relatively expensive method and requiring intravenous contrast administration, dynamic contrast-enhanced (DCE) MRI is the most sensitive modality for any type of breast cancer compared to DM, digital breast tomosynthesis (DBT) and ultrasound with similar specificity(Lehman and Schnall, 2005). According to Mann et al. (2019b), the sensitivity ranges between 81% and 100% in women with various risk profiles, which is approximately twice as high as the sensitivity of DM. Aside from that, MRI is very effective in detecting more aggressive and invasive tumors, which are more relevant in the clinical field. The high sensi-

tivity is due to the fact that no breast cancer can grow larger than 2 mm without forming new blood vessels as the tumor needs to get enough nutrition to develop. (Mann et al., 2019b).

The interpretation of a 4D dimensional volume as a DCE-MRI scan is time-consuming and challenging. With an increasing number of patients undergoing breast MRI, computer-aided detection (CADe) systems that support the radiologist by decreasing interpretation time and oversight mistakes are required (Witowski et al., 2022). As stated by Yamaguchi et al. (2013), in the small dataset used in the study, more than half of the MRI detected cancers could be detected on prior MRI. Despite the significant impact of artificial intelligence in breast cancer detection on DBT and DM, there is no deployment in the clinical practice of an autonomous breast cancer MRI detection pipeline to our knowledge.

1.1. Paper description

Our goal is to propose an innovative pipeline for imaging interpretation of breast MRI examinations that could increase safety and reliability in the future. This means that the suggested method could be the basis for an AI system that positively impacts the breast MRI clinical practice. Our vision is to assist radiologists during their examinations, by reducing the workload and increasing the overall detection performance.

This study proposes a complete deep learning pipeline of malignant lesion detection in non-fat suppressed MRI, exploring both the DCE volumes and extra sequences (T2 and ADC). As shown in Figure 2, the pipeline is divided into three stages: segmentation, candidate detection and classification. For the first part, the well-known 3D U-Net (Ronneberger et al., 2015) and the 3D UNETR (Hatamizadeh et al., 2022) were trained in a patch based approach and compared. Having the breast segmentations, the volumes can be masked and cropped, allowing less computational costs for the rest of the pipeline and reducing false positives by removing detections outside the breast. The 3D Retina-Unet (Jaeger et al., 2020) architecture was used for the candidate detection stage. Finally, in our

1- Segmentation

2- Detection

3- Classification

Input: T1 pre-contrast image
Output: FGT and fatty tissue

for each breast

Input: Non registered subtraction between first T1 post and T1
pre-contrast, masked and cropped leaving only the breasts

Output: bounding boxes for lesion candidates (high sensitivity)

Input: Candidate lesion with context
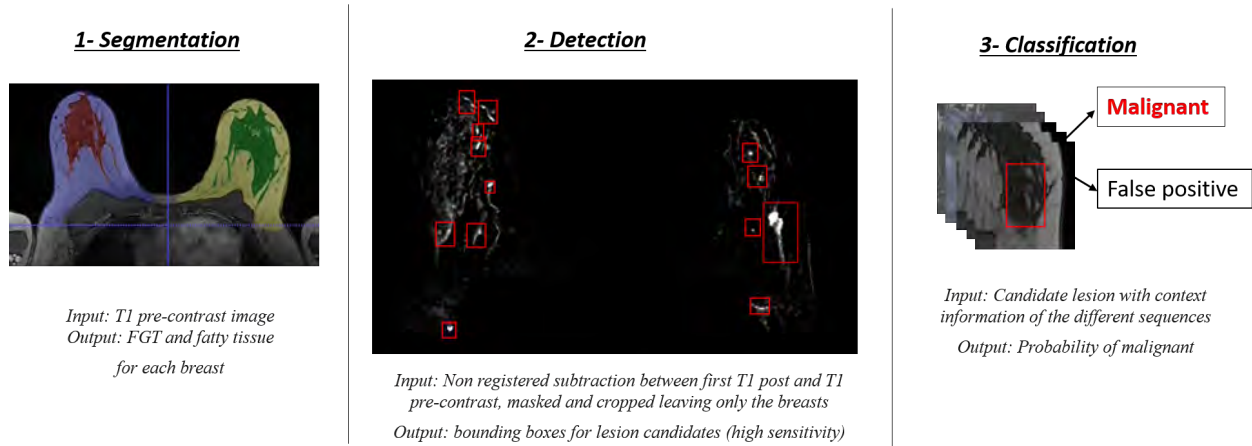information of the different sequences

Output: Probability of malignant

Figure 2: General overview of three stage pipeline: segmentation, candidate lesion detection and classification of each candidate.

classification method, we implemented a 3D Resnet18 to classify each candidate lesion as either malignant or non-malignant, and to examine the impact of the different modalities in the AUROC. Also, first registering the volumes is considered as a way of correcting patient motion during the examination. In addition, we made a different attempt to tackle this problem by using a Vision Transformer (Dosovitskiy et al., 2020) with a self-supervised pre-training method.

This paper is organized as follows. In Section 2, the current scientific work on breast MRI is summarized. The dataset and the networks used in each stage are described in detail in Section 3. Section 4 and 5 present the results and discussion, respectively. Finally, Section 6 summarizes the work.

2. State of the art

Because of the high dimensionality, the variety of protocols and the motion artifacts, automated breast cancer diagnosis in DCE-MRI is a difficult challenge and an active area of research. Our work is based on discoveries in the fields of segmentation, detection and classification. Unfortunately, there is no benchmark for comparing the best performing methods due to the lack of a public complete dataset that includes normal studies, and the results vary considerably between the different private datasets. This section describes the state of the art for segmentation, detection and classification techniques used in breast DCE-MRI.

Gubern-Merida et al. (2014) segmented the breast by automatically detecting body-breast and air-breast surfaces, then FGT was segmented using expectation-maximization. Dalmış et al. (2017) used a 2D U-net to segment fatty tissue and FGT on axial MRI slices. A 2D U-Net++ was implemented by Jiao et al. (2020) to generate breast masks. Due to lower memory requirements, 2D networks can benefit from a larger receptive field. However, as they do not use 3D information, it is more likely to have inconsistencies between different slices in the segmentation. In a recent publication, a plug-and-play tool for the state-of-the-art biomedical segmentation called 3D nn-Unet has been widely used. Huo et al. (2021) implemented a two-stage approach with 2 nn-Unet architectures: first segmenting the whole breast (dice: 0.968) and secondly using the masked volume to segment the FGT(dice: 0.877). Moreover, a single 3D nn-Unet with 3 classes was employed by Samperna et al. (2022) obtaining dice scores of 0.96 and 0.92 for the whole breast and FGT respectively.

Regarding MRI lesion detection, the existing methods fall into two main categories: segmentation or bounding box prediction (or a fusion of them). Dalmış et al. (2018) combined a 2D U-Net to generate a lesion likelihood map with a local maxima algorithm to generate candidates followed by a classification obtaining a CPM of 0.64, and Vidal et al. (2022) implemented a 3D U-Net Ensemble with the same purpose. Zhang et al. (2020) exploited a Mask R-CNN which outputs the bounding boxes and the segmented tumors, and for ultrafast DCE-MRI a modified 3D RetinaNet model that operates on T1w sequences was implemented by Ayatollahi et al. (2021) leading to a CPM of 0.86.

Lastly, classification has been applied locally or globally. One example of the first would be the work of Dalmış et al. (2018), who classified the subtraction candidates extracted from the detection also taking into account the symmetry information of the contralateral breast. Recently, Witowski et al. (2022) implemented a ResNet18 to make a global classification per breast in fat-suppressed studies, the input volume was the concatenation of the pre-contrast and 2 post-contrast series on the channel direction of the MRI study. They achieved an AUROC of 0.920 on their internal dataset for cancer diagnosis and generalized well on other fat-suppressed datasets, claiming that it is possible to reduce benign biopsies by 20%.

Despite the multiple efforts in the segmentation, de-

tection and classification fields for breast MRI, to our knowledge there is no current application of such a pipeline in clinical practice. Our goal is to design an autonomous CADe that can set the basis of a commercial product that works with radiologists to decrease false positives and further increase sensitivity.

3. Material and methods

3.1. Dataset

The dataset used for this study was provided by Radboud University Medical Center (UMC) Nijmegen. After a manual inspection, we removed 6 cases due to apparent errors in the annotations. The final dataset includes 570 studies of a total of 454 women screened between 2011 and 2013 due to intermediate or high risk of developing breast cancer. As shown in Table 1, it consists of 173 normal and 397 abnormal scans from 77 and 377 patients, respectively.

The abnormal cases have a histologically sampled (malignant or benign) finding in their breast MRI scan or, based on MRI follow-up, have a lesion with no visible growth for at least one year later, which is considered benign. Inflamed cysts, lymph nodes, fibroadenomas, adenosis and hamartoma are among the benign lesion types included. The annotated benign lesions were the ones consider difficult by radiologists, which were cause of biopsy or follow-up. In other words, there are exams with easily recognized benign lesions that were not annotated. Due to this limitation of the training data, our proposed method does not detect benign lesions and concentrates on malignant tumors. The dataset has bounding box annotations for each lesion as sets of two 3D coordinates, but does not include information about BIRADS score and pathological subtype, it only has the classification between benign and malignant. The number of lesions per exam varies from one to five and a study can have malignant and benign lesions at the same time. Except for a few cases, these studies contain a T1 weighted (T1w) pre-contrast, at least three T1w post-contrast, a T2w and an ADC volume. For the pre-contrast abnormal volumes, we have breast tissue segmentations of FGT and fatty tissue coming from a teacher network (3D nn-UNet) trained at Radboud UMC (Samperna et al., 2022).

Besides these cases, there are 173 studies that were considered normal, meaning no benign or malignant lesions were found during the examination and based on MRI follow-up. For these scans, the segmentations, the T2 and the ADC volumes are not available. In general, these studies contain a T1w pre-contrast and at least three T1w post-contrast.

All DCE-MRI volumes are axial non fat-saturated gradient echo T1-weighted sequences. The studies were performed using a 3T MR Siemens magneton Trio/Skyra scanner with a 16-channel breast coil. The acquisition parameters were $TE = 1.71ms$, $TR = 5.5ms$ and flip angle of 20. Voxel spacing for the DCE volumes is 0.8 mm×0.8 mm×1 mm (axial direction) and volume sizes are either 448×448×160 or 448×448×176 voxels respectively. The volume size of ADC and T2 varies but was reshaped to match the DCE spacing. An experienced breast radiologist supervised the annotation process, and other breast imaging exams, radiological, and histological reports were available for manually annotating 3D bounding boxes.

The 377 patients with abnormal cases were split in training, validation and testing, following a 60, 20 and 20 percent random split. Those extra cases in which a patient has multiple studies were added to the group in which that patient belongs, resulting in 239/77/81 studies respectively. This was done to avoid having the same patient in two different sets. The data split was respected for the segmentation, the candidate detection and the classification steps. The validation set is used to choose the best checkpoint after training. For the normal patients, a random scan split 66/26/81 was done patient-wise for the classification step. As in the candidate detection stage the focus was achieving high sensitivity, the normal scans were not used for training but all of them were part of the test set for plotting the FROC curve.

	Patients	Studies	Tr.	Val.	Test.	T2/ADC
Abnormals	377	397	239	77	81	Yes
Normals	77	173	66	26	81	No

Table 1: Summary of the internal dataset and the split in training, validation and testing.

3.1.1. Manually generated segmentation dataset

At the final stage of the development of this project, we received from Radboud UMC an unstructured MRI dataset of pre-contrast images with manual segmentations of FGT and fatty tissue. After preparing the data, we ended up with 73 cases in total originated from more than one protocol and different annotators. Also, the spacing and size of the volumes are not consistent. Some volumes are complete, while others were cropped around the breast zone. This dataset is only used for the evaluation of our segmentation network on other distributions of non fat-saturated MRI.

3.2. Method

In order to simplify the process of automatically detecting breast cancer in DCE-MRI, we divided it into three steps as presented in Figure 2. In summary, we segment the breast tissue, detect candidates of malignant lesions and finally classify each detection individually. This allows to have a better understanding of how the implementation works, improve or modify steps independently and use different input information according to the needs of each section.

3.2.1. Tissue segmentation

The first stage of the pipeline consists of the segmentation of the FGT and the fatty tissue from the pre-contrast images. The goal is to be able to focus the detection and classification part on the breast area and be able to output the breast density. For this task, we compared the performance of a well-known 3D U-Net (Ronneberger et al., 2015) with a UNETR architecture developed by Hatamizadeh et al. (2022), who demonstrated that UNETR has a better capability of learning long-range dependencies than vanilla U-Net architectures, by validating its effectiveness on different volumetric segmentation tasks in CT and MRI modalities.

We used a standard 3D U-Net with skip connections of 5 levels of depth with feature maps of 32, 64, 128, 256 and 512. On the encoder, we applied 2 convolutions per level with $kernel\_size = (3 \times 3 \times 3)$, followed by one max pooling with $kernel\_size = (2 \times 2 \times 2)$ and stride of 2 to downsample the features at each level. For the decoder, we applied 2 convolutions per level with $kernel\_size = (3 \times 3 \times 3)$ and one transpose convolution with $kernel\_size = (2 \times 2 \times 2)$ and stride of 2 to upsample the feature map. Instance normalization, dropout of 0,2 and LeakyRelu as activation function were also used. The explained architecture has a total of 22M parameters.

Alternatively, as depicted in the Figures 3 and 4, UNETR is a more complex architecture with 96M parameters that utilizes a transformer as encoder to learn sequence representations of the input volume and effectively capture the global multi-scale information, while also following the successful "U-shaped" network design for the encoder and decoder. The transformer encoder is directly connected to a decoder via skip connections at different resolutions to compute the final semantic segmentation output. We used the default implementation from the authors taking advantage of the CT pre-trained weights from the BTCV challenge. These weights were transferred to our network, just leaving random weights in the last output layers. A dropout of 0.2 is used for the encoder Vision Transformer. To our knowledge, this is the first time a transformer network is applied to breast MRI.

We followed the same training and inference procedure for both architectures. We used a weighted average between cross-entropy and dice score as loss function. Cross entropy has better properties for the gradients which makes the training more stable, while dice is the actual goal of our segmentation. After hyperparameter tuning, we used for both architectures a learning rate of 10^{-4} and weight decay of 10^{-5}. We trained with the pre-contrast volumes on three classes: background, fatty tissue and FGT. As we were also interested in the whole breast mask, we easily calculated it using the union between the fatty tissue and FGT. For a measure of breast density for each breast and for both, we
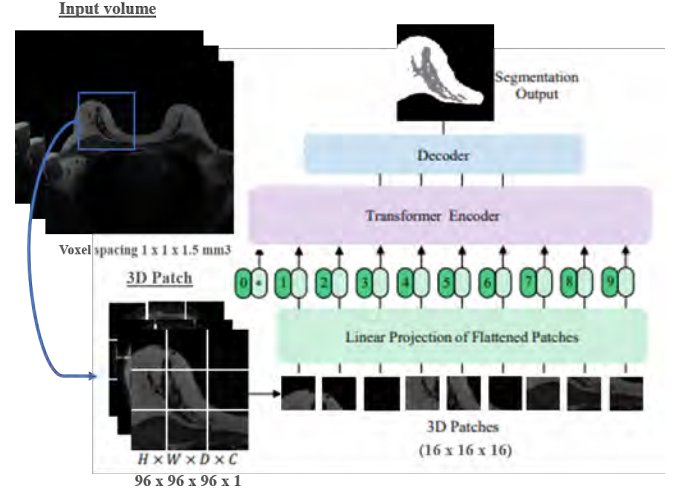
Figure 3: General overview of UNETR. After the image is resized, the network is trained with patches of $96 \times 96 \times 96$ that are divided into subpatches of $16 \times 16 \times 16$ which are the tokens of the transformer projected into an embedding space.

resampled the image to 0.8 mm isotropic spacing with nearest-neighbor approximation, counted the number of voxels per class and computed the breast density as

$$Breast\ Density = \frac{\#FGT\ voxels}{\#FGT + \#Fatty\ tissue} \quad (1)$$

As a normalization technique, values from 0 to 320 were mapped to a range from 0 to 1 without clipping. The values 0 to 320 were chosen after inspecting the images and the range between 0 to 1 was selected in order to use the UNETR pre-trained weights from the BTCV challenge.

The 3D pre-contrast abnormal volumes were resampled in order to have $1mm \times 1mm$ resolution in the axial plane and 1.5mm in the axial axis. Therefore, input images were downsampled to increase the receptive field of the network without increasing the computational complexity. Rotation, flipping and intensity augmentations (contrast adjustment, scaling and shifting) were used with a probability of 0.3 each. Both networks were trained with random patches of $96 \times 96 \times 96$ and a sliding window approach was used at inference time. We decided to apply overlapping windows (Figure 5) and compute a weighted average of the predictions giving more importance to the voxels in the center of the patch than in the borders (following a Gaussian weighting). The overlap is a hyper-parameter for inference time that indicates the amount of overlap between patches for the sliding window approach.

A post-processing pipeline was implemented to refine the segmentation results. It includes filling operation, separating both breasts, finding the two main connected components, reshaping and saving the images. The prediction labels are:
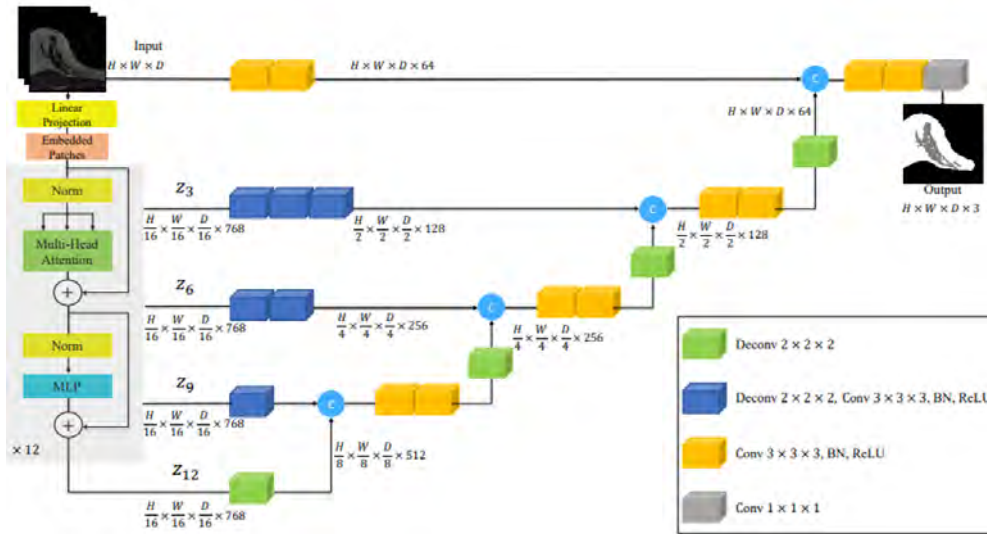
Figure 4: Architecture of UNETR. It resembles the "U-shaped" of U-Net with the encoder and decoder. In U-Net the encoder is convolutional, while UNETR uses a fully transformer encoder whose output tokens are reshaped with deconvolution layers.
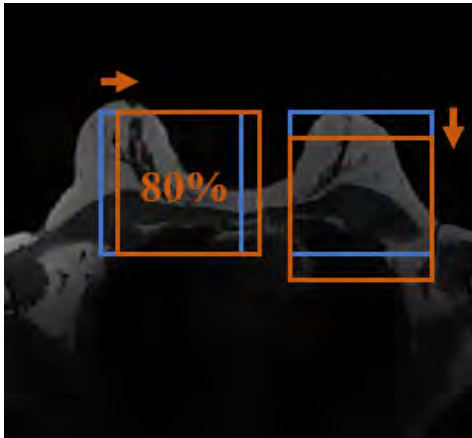
Figure 5: Sliding window approach with overlapping patches (i.e. 80%) for inference time. Predictions are averaged giving more weight to the voxels closer to the center of the patch.

- Background: 0

- Right breast FGT: 1

- Left breast FGT: 2

- Right breast fatty tissue: 3

- Left breast fatty tissue: 4

It is worth mentioning that the ground truth segmentations coming from the teacher network did not use this post-processing. Therefore, the breasts are not separated between right and left breast and can contain more or less than two connected components.

3.2.2. Candidate detection

The main goal of this stage is to find candidate lesions with high sensitivity, considering that the final classifi-

cation step works as a false positive reduction method. This subsection explains how the dataset was utilized, the necessary improvements to the bounding box annotations, the architecture employed, and the outputs.

Input volume and normalization

In breast cancer in MRI, the relative enhancement volume is defined as

$$Relative\ enhancement\ volume = \frac{Post - Pre}{Pre + \varepsilon} \ . \quad (2)$$

Post stands for the first non-registered post-contrast volume and *Pre* for the pre-contrast volume and the variable $\varepsilon = 10^{-7}$ is for numerical stability. The relative enhancement volume is usually used for finding lesions, and in this case, it is masked and cropped with the previously obtained segmentation mask of the breasts. In order to avoid skin artifacts, the segmentation mask is eroded with a circular structural element with radius of 2.

The detection approach is trained on the relative enhancement volumes of abnormal cases to detect all candidate lesions, regardless of being malignant or benign. As it is known that more information than a subtraction volume might be needed to differentiate between these two types of candidates, both of them are included in a single positive class at this stage. This decision was made to increase the sensitivity of malignant lesions. For this stage, normal cases are only used as a test set to calculate the performance.

Annotation's variability

A drawback of our dataset is the high intra and inter variability of the bounding boxes. As seen in Figure 6, in some cases, the annotations seem to be larger than the

Figure 6: Examples of bounding box annotations. In some cases, they are much bigger than the lesion or contain several lesions.
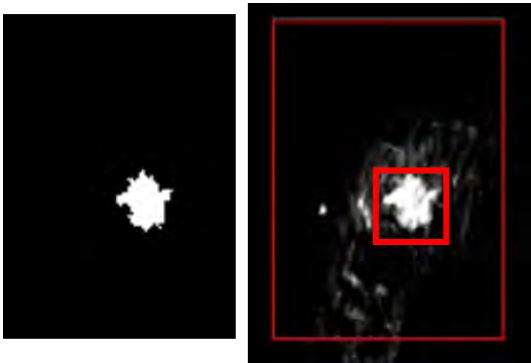
Figure 7: Left to right: result of automatic segmentation of the lesion and final bounding box used for training.

lesion or they contain multiple lesions in one bounding box. Therefore, it is not logical to expect the network to return similar bounding boxes.

To reduce this variability, we decided to automatically perform segmentations from the annotations and generate new bounding boxes from these masks, this is illustrated in Figure 7. The traditional method consists in segmenting the tissue whose intensity values have increased more than 100% (over 1 in the relative enhancement) after the administration of the contrast agent. If the segmented volume is smaller than a certain threshold, we reduce the intensity increment percentage recursively. We remove the small clusters according to 6 connectivity, but if all of them are small, we leave the largest one. We also tried Otsu's method (Otsu, 1979) and adaptive thresholding, but according to our qualitative interpretation, this simple strategy generates more robust outcomes.

Architecture: Retina-Unet

Following the steps of Dalmış et al. (2018), we retrained our UNETR for lesion segmentation. To obtain high sensitivity, a low threshold should be chosen on the probability map that UNETR outputs, which leads to low precision. Getting bounding boxes with a low number of false positives requires carefully designed handcrafted rules and ad-hoc heuristics when mapping back
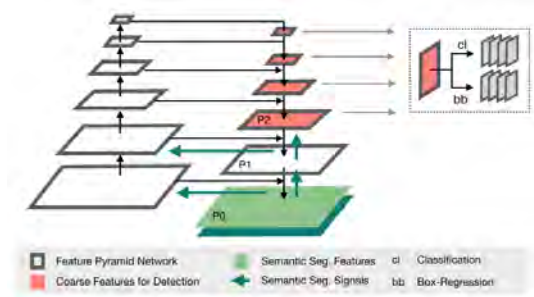
Figure 8: The Retina U-Net architecture in 2D Jaeger et al. (2020). Pj denotes the feature-maps of the jth decoder level, where j increases as the resolution decreases.

to object-level scores. We concluded that it is better to have a trainable approach that can be easily fine-tuned and adaptable to different situations.

One architecture that satisfies our requirements is Retina-Unet developed by Jaeger et al. (2020). It is specifically designed for medical image object detection with the ability to exploit the full pixel-wise supervision signal. Specifically, we benefit from the segmentation maps as an extra training task which in a way compensates for the small dataset available. Hence, the network is trained simultaneously through the segmentation and the bounding boxes.

As Figure 8 describes, Retina U-Net fuses the Retina-Net (Lin et al., 2017) one-stage detector with the U-Net architecture which is widely used for semantic segmentation in medical images. It is based on a Feature Pyramid Network(FPN) with skip connections for feature extraction, where two sub-networks operate on the pyramid levels P2-P6 for classification and bounding box regression, and the top-down part of the FPN with high-resolution levels is connected to the segmentation head. The branches for classification and regression are applied at 4 different scales and they share weights. For the classification head, it uses online hard negative mining with cross-entropy (CE), for the segmentation $(CE_{loss} + Dice_{loss})/2$ and for the bounding box regression $L1$. The total loss is the sum of these terms.

Retina-Unet is not maintained anymore by the authors as it is included in the framework nnDetection (Baumgartner et al., 2021). This is a self-configuring framework, but the main disadvantage is its extra complexity which makes it difficult to modify the architecture. As we are interested in a more flexible option, we decided to upgrade the original 3D Retina-Unet implementation of the authors by fixing libraries incompatibilities and solving CUDA errors in Non-Maximum Suppression.

For training, all volumes are reshaped to isotropic 0.8 mm spacing in order to work with the highest resolution among the three dimensions in the volume. Rotation and scaling are used as data-augmentation techniques and 3D patches of $96 \times 96 \times 96$ or $128 \times 128 \times 128$ are

Figure 9: Example of the output of NMS and WBC for a cluster of three detections. The final prediction box and score are shown in green.

	Malignants	Benigns	Normals
Training	214	78	317
Validation	81	23	127
Testing	68	32	308
Total	**363**	**133**	**752**

Table 2: Number of lesions per type and dataset split for the classification approach. These are the ones that have at least a pre-contrast and three post-contrasts. For different volume combinations that we determine for training, these numbers can vary. The most significant is that normals do not have T2 or ADC volumes, thus, if we decide to include those, we will only use the abnormal volumes for training. Also there can be some other minor changes in the quantities, for instance, there are a few more examples if we considered cases with at least only one than three post-contrasts.

used for training. Using a ResNet50 backbone for FPN, anchor cubes of 8,16,32,64 with aspect ratios of 0.5, 1, 2 for the corresponding pyramid levels P2, P3, P4, P5. Weight decay is used as a regularization technique.
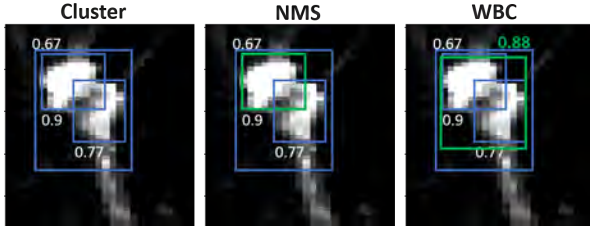
At inference time, overlapping 3D patches are employed in a sliding window approach. Building an ensemble of the different checkpoints and data augmentation at test time are recommended by Jaeger et al. (2020) as a way of improving accuracy and robustness. The best checkpoints for the ensemble are selected according to mean average precision performance in the validation set. Rather than directly using Non-Maximum Suppression (NMS) to obtain the final bounding boxes from the detections, Weighted Box Clustering (WBC) is applied. In this case, instead of selecting the highest scoring box in a cluster as with NMS, WBC computes weighted averages per coordinate and a weighted confidence score to output a final prediction box, which has new coordinates and a new confidence score as in the example in Figure 9. This calculation takes into account the intersection over the union (IoU) with the highest score inside a cluster, the area of each box, gaussian weighting (down-weights prediction boxes close to the borders of the patch), the scores and coordinates of each box. Prior knowledge about the expected number of predictions at a position from the ensemble, test time augmentations, and patch overlap is also considered.

Performance and output

The Free-response Receiver Operating Characteristic (FROC) curve is computed to evaluate the proposed method. This curve displays the sensitivity versus the average number of false positives per normal scan, defining as a true positive when the IoU between the predicted bounding box and the ground truth is higher than 0.1 (default value for 3D medical imaging lesion detection). At exam level, a scan is considered true positive if the network predicts at least one of the lesions of that study. This metric is used to determine the best-performing model. In general abnormal patients of our dataset have only one lesion, the sensitivity per lesion is comparable to the sensitivity per scan. Finally, an operating point with high sensitivity is determined by setting a threshold on the score. The resulting detections are the input of the false positive reduction classification step.

3.2.3. Classification

Throughout this section, we explain the methodology for using different registered or non-registered extra sequences (post-contrasts, T2 and ADC) with ResNet18 for the final classification. Furthermore, we describe the use of Vision Transformers. For the classification stage, three main experiments are carried out:

- The comparison between concatenating volumes with or without registration using ResNet18.

- The additional value of each sequence in a DCE-MRI study in terms of the classification performance using ResNet18.

- The performance of a more flexible approach as Vision Transformer when using non-registered volumes.

Generating input volumes

In the previous step, we detect candidates of malignant and difficult benign lesions (see 3.1) as the positive class. Detections with a score over 0.25 enter the classification stage. As seen in Table 2, we added the labels of malignant, benign or normal for each of them. The annotated benign lesions in studies which also contain malignant lesions are removed. We decided to use only the detections for training instead of using the ground truth patches. The main reason is for the network to receive the same input in training as at inference time. As they are usually similar, we do not expect a significant gain in also adding the ground truth patches at training time.

The detection bounding boxes are on the subtraction volume. Although the volumes are not perfectly aligned due to patient movement during the procedure or differences in the modality acquisition as with diffusion images, we can estimate that the lesion should be found in a similar position. As a result, we crop a window of $64 \times 64 \times 64$ around the center of the detection for each original volume (Figure 10) and concatenate them to form the input volume. It is important to highlight that for the normal exams, the T2 and ADC volumes are

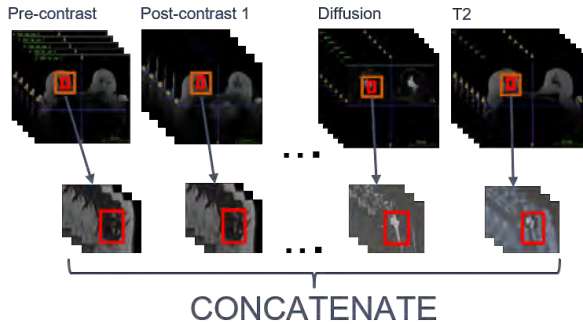Pre-contrast   Post-contrast 1   Diffusion   T2

...

...

CONCATENATE

Figure 10: From the center position of the detection a window of $64 \times 64 \times 64$ is cropped from each volume. The resulting patches are concatenated to generate the input volume.

not available (Table 1). As the bounding boxes output by the network are accurate per lesion, they are usually small and this window size is enough to cover the lesion as well as some contextual information. The idea behind using a fixed window size is to be able to do inference with a constant voxel spacing.

Another approach is to first register the volumes using as the fixed image the pre-contrast and afterward crop the 64 voxels wide windows. We use Elastix (Klein et al., 2009) with an algorithm that combines rigid and non-rigid B-splines transformations in a multi-resolution scheme developed by Gubern-Mérida et al. (2015). Stochastic gradient descent optimizer and mutual information are applied. Grid spacing for the B-splines is set to 160, 80 and 40 mm respectively for the three resolutions. A coarse spacing is applied to reduce the deformation of the lesions.

Each patch is normalized before concatenation. For the DCE sequence, the intensities are divided by the mean of the fatty tissue in the pre-contrast volume. For T2w, the mean of the fatty tissue in the T2w volume is used by applying the same segmentation mask from the T1w pre-contrast (assuming they are perfectly aligned). As the diffusion volume is intrinsically different, we normalize it to the desired range of intensities by dividing by 1400. Because the ADC absolute intensity values are the measure of a physical property, this simple mapping should not affect it.

Output values

In the last layer of both classification architectures, the output of the two neurons goes through a Softmax activation function and the value corresponding to the positive class is interpreted as the probability of malignancy (0 to 1), which is the classification score.

ResNet18

We used the pre-contrast, first post-contrast and last post-contrast to do a quick search for the best CNN architecture for our classification problem. For this, we trained on the false positives detected on normals

and the true positive benign and malignant lesions. We classified them into non-malignant and malignant. The architectures that we examined were DenseNet121, DenseNet201, SEResNet50, SEResNet101, ResNet18 and ResNet50, all of them in 3D. Without doing an in-depth study, ResNet18 proved to be among the best performers, so we decided to continue with this model as in Witowski et al. (2022).

In addition, we implemented the necessary changes in 3D-ResNet18 to match the details describe in their paper. Specifically, the 3D-ResNet18 has a max-pooling layer before the linear classifier, batch norm, batch size of 16, dropout of 15% in the fully connected layers, weight decay as regularization and is trained with Adam optimizer from scratch in all the cases. As data augmentation techniques, we use flipping, random scale and shift intensities, 90 degrees rotations in the axial plane and random crops around the center (not necessarily symmetric) with a minimum size of 48^3 that are afterward resized into 64^3.

For the ResNet18 architecture, the different volumes are concatenated in the channel direction, which means that the input shape is [*channels*, 64, 64, 64] with channels varying from 1 (only pre-contrast) to 6 (pre-contrast, three post-contrasts, T2 and ADC). Basically, different models are trained using various combinations of DCE volumes as inputs. For normal patients the ADC and T2 volume are not available, therefore, a different model is trained to classify only between the benign and malignant lesions using these extra sequences. The idea of training these different models is to evaluate the extra value of adding each sequence.

ResNet18 occlusion sensitivity

To see why the network makes a particular decision, we compute the occlusion sensitivity (Zeiler and Fergus, 2014) on numerous cases from our testing set. Our main goal is to rule out biases in the dataset by verifying that the model is concentrating on the lesion. We occlude part of the image and see how the probability of a given prediction changes. We then iterate over the image, moving the occluded portion as we go, and in doing so we build up a sensitivity map detailing which areas are the most important for that decision. As important parts of the image are occluded, the probability of classifying the image correctly will decrease. Hence, more negative values imply the corresponding occluded volume was more important in the decision process

Vision Transformer

Vision Transformer (Dosovitskiy et al., 2020) is a modified Natural Language Processing transformer for image classification without any convolutional layer. An image is split into patches ($16 \times 16 \times 16$) and put into a lower-dimensional embedding space. Information about the relative position of the patch in the images is added to each vector through positional embedding and a learnable extra token is added to the entire sequence
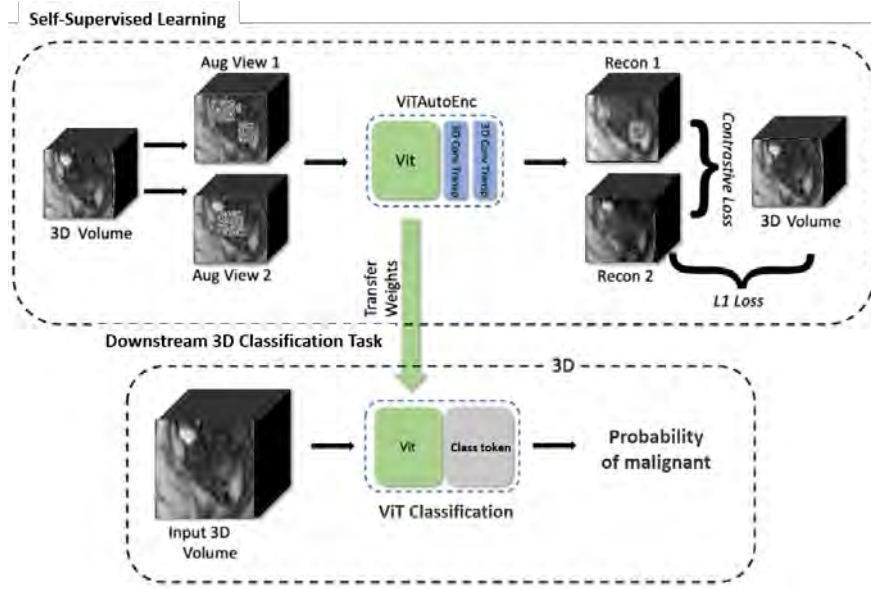
Figure 11: Self Supervised Learning for ViT through proxy task. The learned weights are transferred to the downstream task for the binary classification.

of vectors to denote the class. The sequence of vectors is fed to the standard Transformer encoder, which has been modified with an extra fully-connected layer at the end for classification.

Instead of concatenating the cropped volumes in the channel dimension, we concatenate them in the axial axis. By building relations between all the tokens, the network learns how to relate the different sequences internally. This is due to the flexibility that the attention mechanism in Vision Transformers provides, while CNNs have a more local receptive field. Therefore, this architecture does not need registration which avoids the downsides of this method: higher inference time, lesion deformation and in some cases registration does not conclude in satisfying results.

However, the lack of inductive biases goes along with a higher number of parameters can be challenging to train with small datasets. To address this issue, a Self Supervised Learning (SSL) method as a pre-training task is applied. As indicated in Figure 11, the pre-training pipeline uses augmentations to generate two versions of the volume. There is a ViTAutoEnc that tries to recover the initial volume for each of them. The model is trained through reconstruction and contrastive loss to learn a feature representation of the unlabeled data.

3.2.4. Preliminary automatic report

Combining the three stages, a basic report is generated with the density information, the global score and the detected lesions. The volumetric FGT percentage is displayed for each breast and for both, together with the segmentation mask. For visualization purposes the probabilities of malignancy for each detection are mul-

tiplied by 100 and shown as a number between 0 to 100. The study's global score (0 to 10) is calculated by multiplying the highest bounding box probability of that study by 10.

3.2.5. Statistical analysis

For the segmentation results, the dice scores are reported with the standard deviations. Graphs and tables of detection and classification stages show the results with 90% confidence interval using bootstrapping of 35 samples. To compare if one model is better than the other, Wilcoxon one-tailed paired signed-rank test (Wilcoxon, 1992) is used with a critical p value of 0.05.

3.2.6. Framework

The pipeline was implemented with Pytorch version 1.10.2 and CUDA 11.3 on a Linux environment. Pytorch-Lighting, Tensorboard and MONAI were used with versions 1.5.9, 2.8.0 and 0.8.0 respectively. The trainings were performed on a Nvidia Titan V GPU with 12GB of memory.

4. Results

4.1. Tissue segmentation

As previously stated, we have defined learning rate, weight decay, dropout, loss function and data augmentation techniques for the three class segmentation with UNETR. The U-Net architecture is more robust to these hyperparameters and the same configuration as with UNETR works properly. Experiments were performed with different voxel spacing and patch sizes, concluding that spacing of $1 \times 1 \times 1.5$ (1.5 in the axial axis) and patches of $96 \times 96 \times 96$ achieves a good performance.

15.10
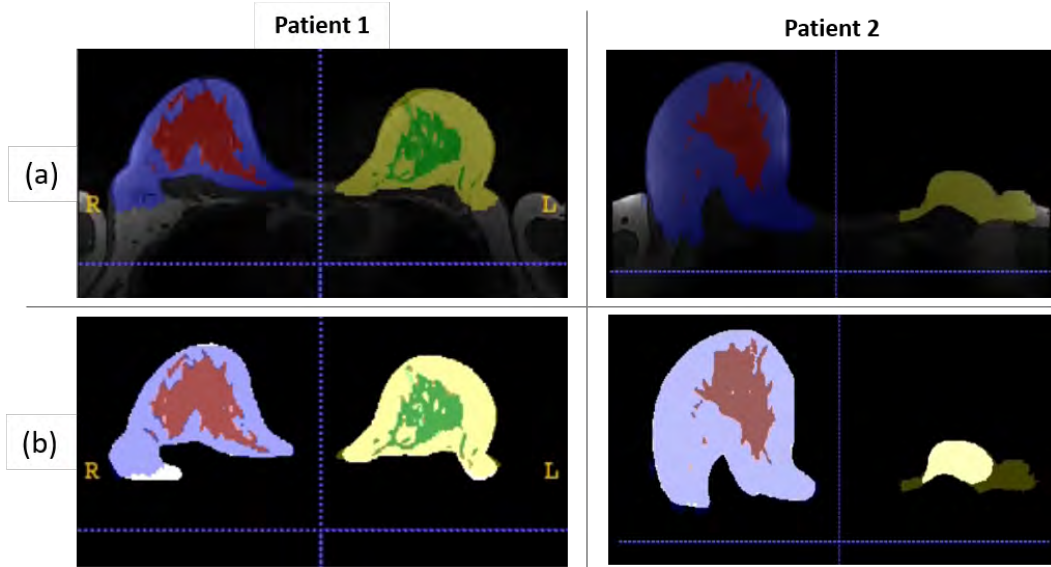
Figure 12: Top to bottom: (a) Predicted segmentation, (b) Prediction and ground truth. In colors, output examples of UNETR using overlapping patches of 80% on the internal dataset, and comparison with ground truth mask in second row. The second example shows the output on a mastectomy patient.

Model ± SD	Overlap	Dice FGT	Dice Fatty	Dice whole breast	Time GPU	Time CPU
UNETR (binary)	0.8	-	-	0.932±0.053	22 sec	620 sec
UNETR	0.8	0.884±0.05	0.940±0.02	0.963±0.014	24 sec	690 sec
UNETR	0.6	0.881±0.05	0.936±0.02	0.960±0.020	11 sec	-
U-Net	0.8	0.877±0.06	0.944±0.02	0.966±0.012	46 sec	1600 sec
U-Net	0.4	0.873±0.07	0.939±0.02	0.964±0.018	10 sec	-

Table 3: Average dice score and standard deviation over the cases of the testing set. The time refers to the total inference time per volume.

Table 3 presents the performance of 5 different models. One model was trained for a binary segmentation between breast and background, resulting in a lower dice score than other models that segment FGT and fatty tissue. As expected, an increase in the overlap slightly improves performance but at the cost of a longer computational time. Although UNETR and U-Net are very different architectures, they perform very similarly. Some UNETR segmentation results can be seen in Figure 12; U-Net segmentation results are not presented because visually they are almost identical. UNETR performs slightly better in FGT, while UNET in fatty tissue. At inference time, UNETR is twice as fast for the same configuration. When running on CPU without any additional optimization, both architectures show a large increase in computing time (see Table 3).

Finally, U-Net has the benefit that can be applied to different input image sizes. Although we trained with patches, we experiment with predicting the whole volume in one forward pass. However, because to GPU memory limits, we decided to do inference with overlapping large patches of $256 \times 256 \times 96$ under the same conditions. As a result, the inference is slightly faster but the dice score decreased by about 1% for whole breast, FGT and fatty tissue.

Evaluating on manually generated dataset

Besides the internal testing split, we evaluate the previous UNETR and U-Net with a sliding window overlap of 0.8 with the external manual segmentation dataset. The results are summarized in Table 4. Evidently, there is a performance drop for this dataset. The standard deviation is much higher, meaning that there is more variability in the performance.

In Figure 13, we show two output predictions of UNETR and U-Net. For example a) both networks receives low scores, but being lower for the UNETR. In the alternative case, both architectures achieve good segmentation results in terms of dice score.

Model	Dice FGT	Dice Fatty	Dice breast
UNETR	0.68±0.16	0.82±0.14	0.86±0.1
U-Net	0.71±0.16	0.83±0.13	0.88±0.07

Table 4: Average dice score and standard deviation over the cases of the external manual evaluation set.

U-Net   UNETR

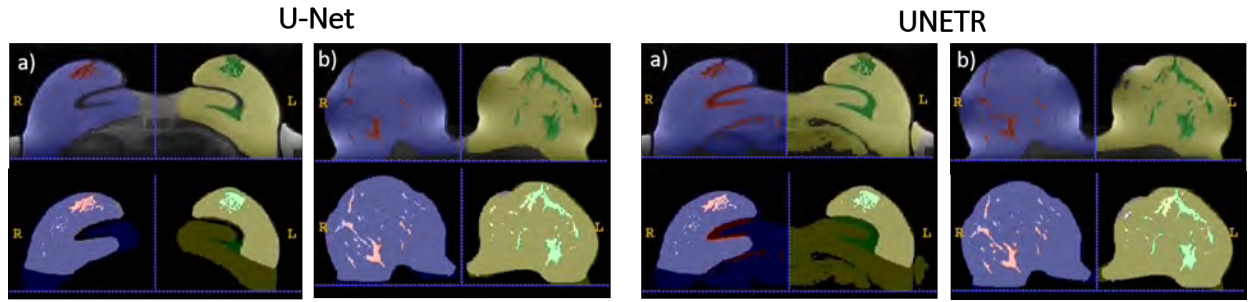Figure 13: Output examples of two patients from the external manual dataset with UNET and UNETR (0.8 patch overlap). First row presents the output of our networks in colors and the second row overlaps the predictions with the ground truth. The example a) is a low dice score while b) is a high dice score result.

4.2. Candidate detection

Once Retina-Unet was implemented, some issues arose at training the model from scratch. The training loss was oscillating and not decreasing significantly. Different hyper-parameters did not solve the problem, therefore, we decided to start training from the segmentation, similar to a U-Net and, later, added the classification and regression heads. When training only from the segmentation signal, the loss decreased consistently. The feature maps were already representative, which made the training of the classification and regression heads easier. Usually, pre-trained weights are used for detection networks, but they were not available for the 3D FPN.

To decide which was our best performing model, we performed 4 experiments:

1. Model of patches of 96^3 and inference with one single checkpoint.
2. Model of patches of 96^3 and inference with an ensemble of the best three checkpoints.
3. Model of patches of 128^3 and inference without data augmentation.
4. Model of patches of 128^3 and inference with four data augmentations per patch. This means flipping the patch in three different ways, predicting and, as usual, the findings were gathered using Weighted Box Clustering.

Although the positive class consists of benign and malignant lesions, we calculate sensitivity on malignants as it is our main focus. The FROC curves of the models are presented in Figure 14. It is important to take into consideration that using three checkpoints or four data augmentation triples or quadruples the inference time, respectively. Therefore, we decided to use the model with a patch size of 96, with only the best checkpoint and without test time augmentations.

Figure 15 illustrates the output of the detection network with their respective score values. It can have very high sensitivity for malignant lesions, but it also finds many suspicious lesions in normal patients. This is expected as the network was trained only with images of
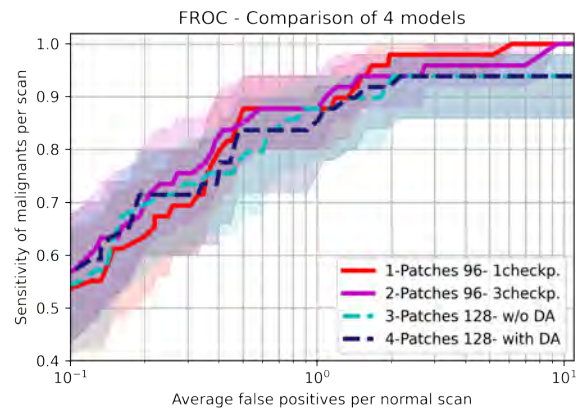
Figure 14: FROC of candidate detection models for different patch sizes, number of checkpoints for ensemble and with or without data augmentation.

abnormalities, including benign lesions in the positive class.

We observed that some specific cases have many detections (up to 20) with most of them being false positives. Thus, we performed some experiments to observe the changes in the FROC curve when limiting the number of detections per scan to the top 1, 3, 5 or 10. Although a limit to three detections generally improves the results, it only happens when the sensitivity is lower than 94%. As we are interested in even higher sensitivity values to input to the classification network, we decided not to include a threshold for the number of detections.

As previously stated, this stage of the pipeline is a candidate detector that outputs a score for each detection. To determine the input to the classification stage, we decided to use a threshold of 0.25 on the score. According to the selected model in Figure 16, the average false positive per normal exam is 5 and the sensitivity is over 98%.

Abnormal patients Abnormal MIP Normal patients
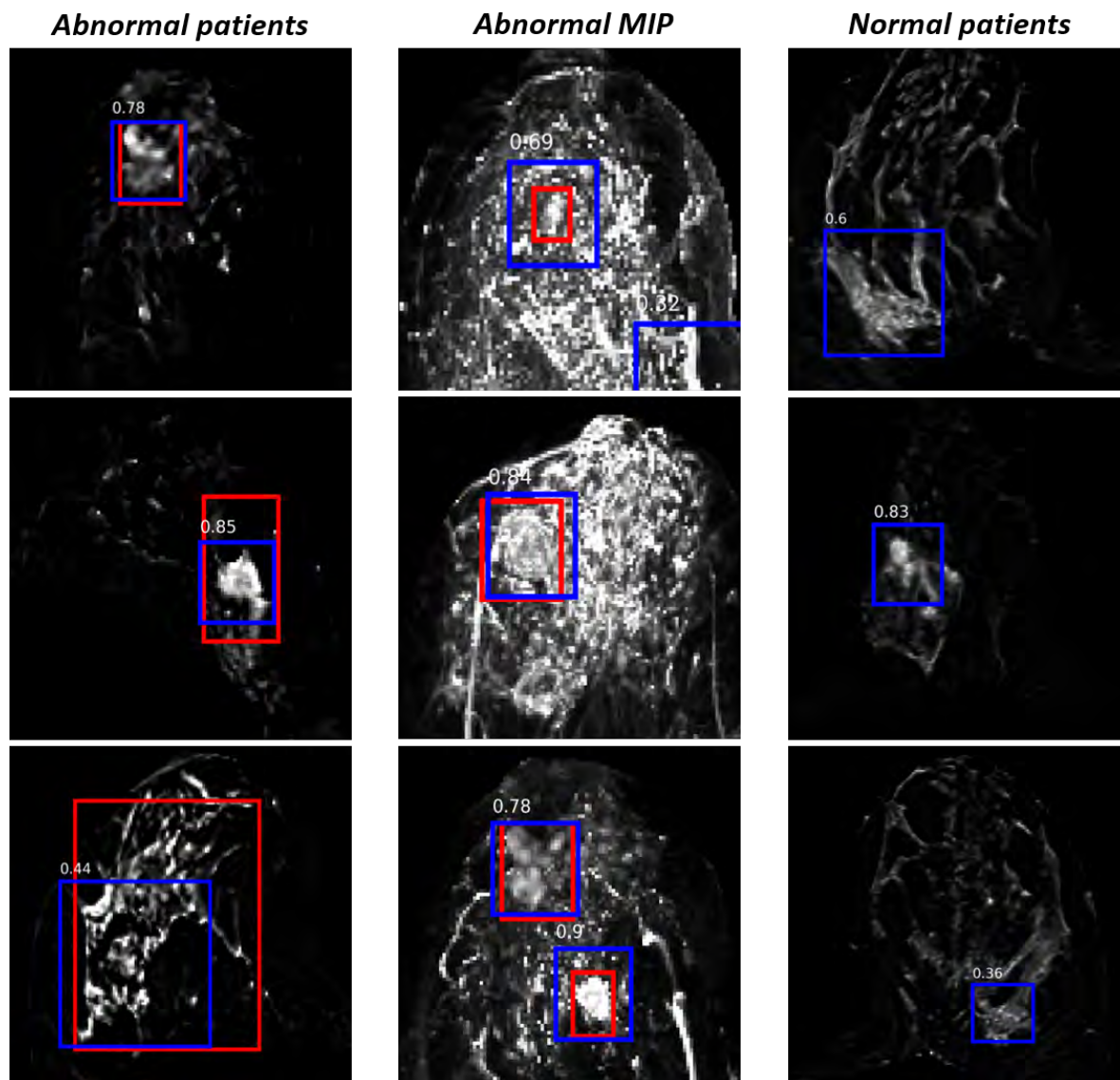
Figure 15: Examples of the predictions in blue of the final candidate detection network in abnormal and normal images. Red boxes refer to annotated benign or malignant lesions. MIP is the maximum intensity projection in the axial axis.
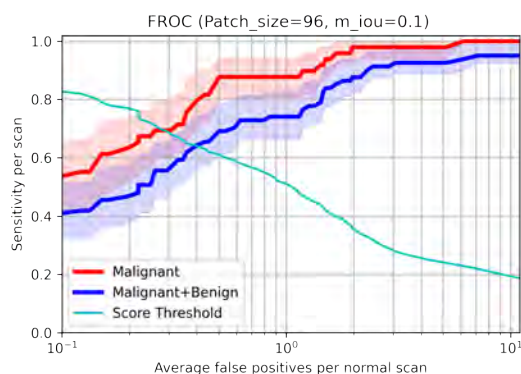
Figure 16: FROC for final candidate detection model. In red the sensitivity in malignant scans with CPM=0.83 (0.73-0.89), and in blue, the overall sensitivity for malignant and benign.

4.3. Classification

4.3.1. ResNet18

DCE sequence

To determine the additional value of each volume in the DCE sequence, the model's performance when each volume is added as an extra channel was compared. For this binary classification, the models were trained on malignants, benigns and normals, oversampling to compensate for the imbalance by duplicating the malignants and triplicating the benigns. The AUC for the testing set can be seen in Figure 17 compared with the models trained in the same way but on registered images.
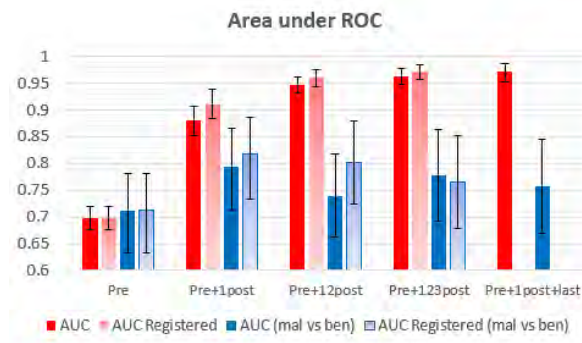
Figure 17: Models trained with normal, benign and malignant lesions for different volumes of DCE sequence. The chart summarizes the results of 8 models. Area under ROC for models based of unregistered volumes are shown in red and blue, and for registered volumes as input in light colors. AUC refers to malignants versus benigns and normals, whereas the AUC between malignants and benigns corresponds to the same model but excluding the normals of the analysis. Pre stands for pre-contrast while post with the numbers stands for the post-contrasts sequences used. For instance, Pre+12post corresponds to using the pre-contrast and the first and second post-contrast volumes concatenated in the channel direction.

Extra value of T2 and ADC

For this experiment, we conducted a benign versus malignant classification since only the abnormal scans of our dataset have a T2 and ADC volume. Figure 18 displays the performance and the comparison with using registration. The difference between the blue bars in Figure 17 and the ones in this Figure relies on a different training set, as normals are not used for training this last one.
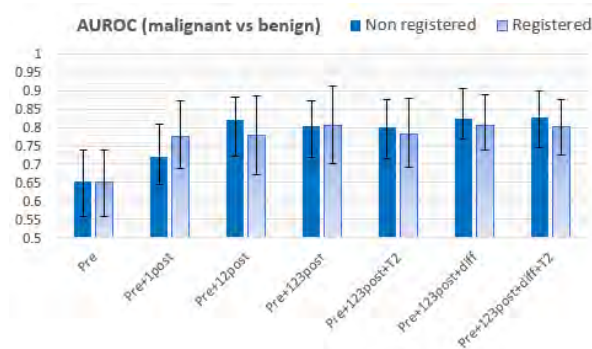
Figure 18: Area under ROC for models with different input sequences using only benign and malignant cases. Pre stands for pre-contrast while post with the numbers stands for the post-contrasts sequences used.

Occlusion sensitivity

Figures 19 and 20 present four examples of occlusion sensitivity maps. The first corresponds to benign lesions classified in the negative class (non-malignant) and the second one to malignant lesions of the positive class. For the malignant cases, the probability of predicting malignancy decrease when the lesion is occluded, whereas for the benign ones the probability of classifying as non-malignant increase by occluding the lesion.
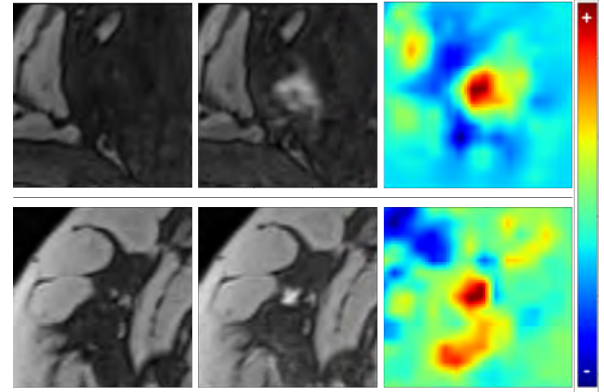
Figure 19: Occlusion sensitivity for two benign lesions of different patients using ResNet18 on Pre+123post input volume with mask size of 8 and stride of 4. Left to right: pre-contrasts, first post-contrast and occlusion map.
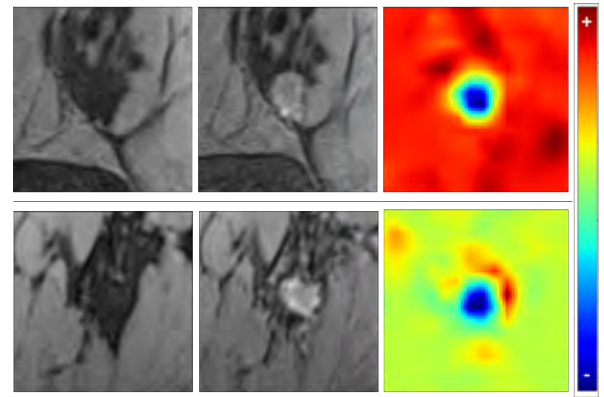
Figure 20: Occlusion sensitivity for two malignant lesions of different patients using ResNet18 on Pre+123post input volume with mask size of 8 and stride of 4. Left to right: pre-contrasts, first post-contrast and occlusion map.

4.3.2. Vision transformer

In order to compare the performance with ResNet18, we trained a ViT with the same input volumes by concatenating, in the channel direction, the pre-contrast and the first two post-contrasts. As shown in Table 5, changing the patch size from 16^3 to 8^3 resulted in a decrease in AUC (malignant versus benign and normal) but an increase in AUC of malignant versus benign. Another modification was to pre-train the ViT on a proxy task in a self-supervised approach by using a contrastive and reconstruction loss function as explained in 3.2.3. This led to an improvement in performance, and a more stable and shorter training time. This model does not perform as well as ResNet18 for classification between malignants and benigns, but it performs similarly if the normals are included in the negative class for the analysis.

The main goal of using a ViT is to provide the network with enough flexibility to correct motion misalignments. It is evident that by adding the extra sequences as channels, this is not possible. Therefore, by concatenating in the axial direction (only 1 channel), it should be able to learn the long-distance relations between the different post-contrasts. The abrupt decrease in performance of d) in table 5 indicates that this is not the case. As with the previous experiments of having three channels, the pre-training helped to simplify the training but the results were still not satisfactory.

Model	AUC	AUC (mal vs ben)
a- PS:16 Ch.:3	0.94(0.92-0.96)	0.66(0.57-0.75)
b- PS:16 Ch.:3 SSL	0.96(0.95-0.97)	0.67(0.57-0.75)
c- PS:8 Ch.:3	0.93(0.91-0.96)	0.71(0.63-0.79)
d- PS:16 Ch.:1	0.66(0.60-0.72)	0.70(0.62-0.78)
e- PS:16 Ch.:1 SSL	0.74(0.68-0.78)	0.65(0.54-0.76)

Table 5: Results for different Vision Transformer models. PS: patch size, Ch:Number of channels, SSL: with Self Supervised Learning as pre-training.

4.3.3. Final classification model

Following the analysis of the previously mentioned models, we selected the ResNet18 trained on pre-contrast and the first three post-contrasts (Pre+123post) without registration for our final pipeline. As seen in Figure 17, it outperforms models using less post-contrasts ($p\ value<$ 0.001, Wilcoxon test with H_1:$AUC_{Pre+1post}$ < $AUC_{Pre+123post}$). Although there is a significant improvement with the use of registration for the model Pre+123post ($p\ value=$ 0.044, Wilcoxon test with H_1:$AUC_{w/o\ reg.}$ < $AUC_{with\ reg.}$), we decided not to include registration in the final pipeline due to the extra computational cost. The ROC curves for this final classification model are displayed in Figure 21.
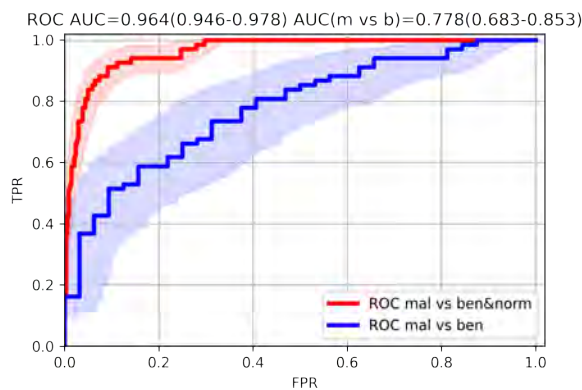
Figure 21: ROC curve of the final classification stage using ResNet18 on non-registered Pre+123post.

The detection network score implies the probability of being benign or malignant, while the output of the classification is the probability of malignancy. Figure
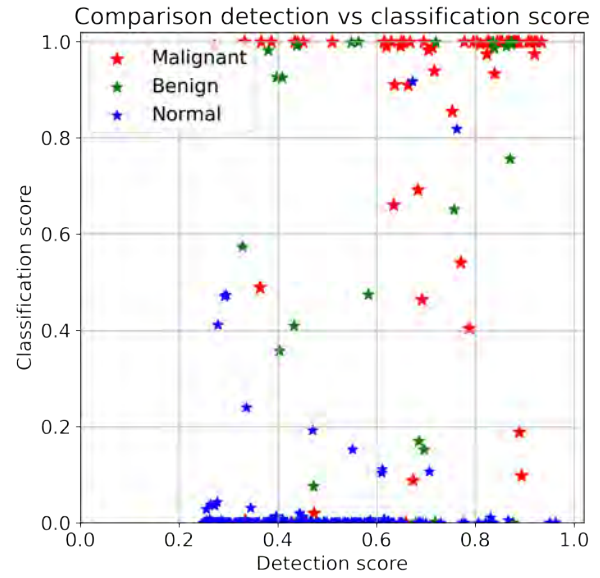
Figure 22: Comparison of the candidate detection scores of Retina-Unet with the predicted classification scores using ResNet18 on non-registered Pre+123post.

22 relates these scores for the different types of lesions of the testing set.

By combining the candidate detection and classification model, it is possible to plot the FROC for the whole pipeline as shown in Figure 23. There is a clear performance improvement due to the reduction in false positives which shifts the curve to the left part of the plot.

4.4. Final pipeline and preliminary automatic MRI report

The diagram in Figure 24 summarizes the models used in the pipeline as a result of the experiments. By doing inference with this pipeline, preliminary auto-
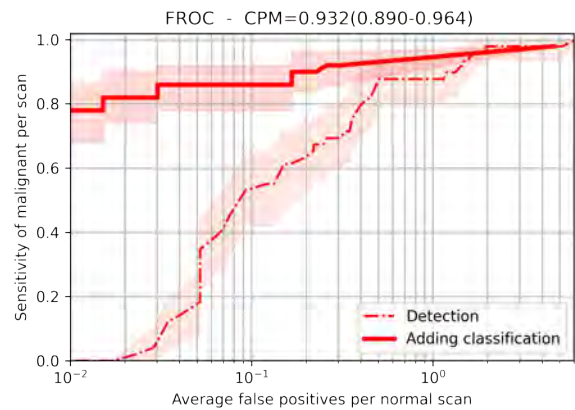
Figure 23: Final FROC curve of the whole pipeline showing importance of the classification stage for the reduction of false positives.
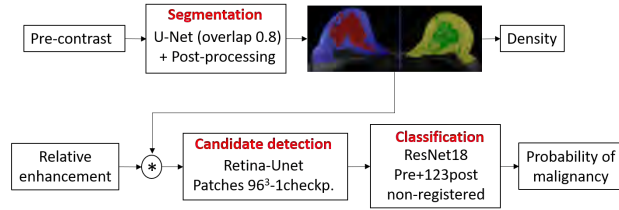
Figure 24: Diagram of the models used in the final pipeline.

matic reports are generated and an example for one scan is shown in Figure 25. The patient information with the density values, the exam score (0-10) and the segmentation mask is shown in the upper part of the report. We decided to include an axial MIP of the relative enhancement volume with all the detections whose probabilities of malignancy are over 5%. In addition, each detected lesion is marked on its center slice on the subtraction volume along with the pre-contrast and the three post-contrast sequences.

5. Discussion

In this study, we developed an automatic CADe system capable of detecting breast cancer lesions in DCE-MRI. To accomplish this purpose, the project was divided into three major tasks. First, segmenting FGT and fatty tissue, which allows to concentrate on the breast area and to report the values for density. Followed by a candidate detection approach using Retina-Unet, and lastly, a classification per candidate as a false positive reduction method. For clarity, this section discusses each stage separately.

5.1. Tissue segmentation

For the internal dataset, UNETR and U-Net output similar results, which are also very close to the ground truth. Values of dice score over 0.96 for the whole breast show that the segmentation task can be considered complete for this dataset. It is important to take into account that sometimes there is no clear boundary for the breast on the posterior side and this is the main reason for disagreement between the teacher and the student networks. This can be observed in Figure 12. For the external dataset, this phenomenon is amplified due to the intra and inter variability of the manual annotations. An example of the U-Net architecture is displayed in Figure 13, where the manual ground truth does not include the posterior sector of the breast. For this case, although the output segmentation receives a low dice score, it is debatable if it is indeed a mistake.

As mentioned previously, both networks output similar results for the internal dataset, but are slightly better for U-Net. We hypothesized that this difference is because the teacher network is a U-Net architecture, which makes it easier to mimic its predictions. But this

small difference is amplified with the external manual dataset, which indicates that U-Net generalizes better to unknown data distributions. UNETR has more parameters which make it more likely to overfit the specific training data. Although we used dropout and data augmentation, it did not fully compensate for it.

In general, there is a significant decrease with the external manual annotated dataset. In a certain way, it was expected as the training is performed with a very homogeneous set and data normalization is not robust enough, limiting generalization capabilities. In some cases, the volumes of the external dataset are cropped to the breast area, but we believe this is not a problem for our networks that work with a sliding window patch approach. We assume that the reduction of the dice scores is due to a different intensity distribution and the higher variability in the annotations. By fine-tuning with part of the manual dataset, we can improve the results considerably, but our goal for this experiment was to analyze the generalization capabilities.

In terms of speed, although UNETR has 4 times more parameters, it is twice as fast at inference time. Due to the 3D kernels in the encoder part of the U-Net, which requires a low number of parameters but 3D convolutions are computationally expensive. Linear layers and dot product attention are faster to compute in this scenario. It is possible to decrease the computational time for the U-Net by doing inference with larger patches of $256 \times 256 \times 96$, but resulted in a decrease in performance. We assume that the large receptive field of the CNN normally sees padding around the patches, while with a larger patch size non-zero values are included.

Comparing to the work of Huo et al. (2021) and Samperna et al. (2022) that achieved dice for the breast of 0.968 and 0.96 respectively, we reached a similar performance on our internal dataset. It is important to clarify that we are evaluating on a different dataset from a teacher network predictions, which may contain mistakes and simplifications despite being visually reviewed. Due to the slightly better performance and the superior generalization capability of U-Net architecture, we decided to use this simpler approach with sliding window overlap of 0.8 for the final pipeline.

Concerning the density estimations, we presented them in a quantitative manner as the percentage of fibroglandular tissue in the breast, while radiologists classify the breasts into 4 classes in increasing order density: A, B, C or D. Choosing the correct three thresholds to classify into the different density groups is left for future work. The advantage of relying in a quantitative method instead of the radiologist is that it standardizes the density results, removing the variability of the reader.

5.2. Candidate detection

One of the problems that we encountered was the variability of the bounding box annotations. By seg-
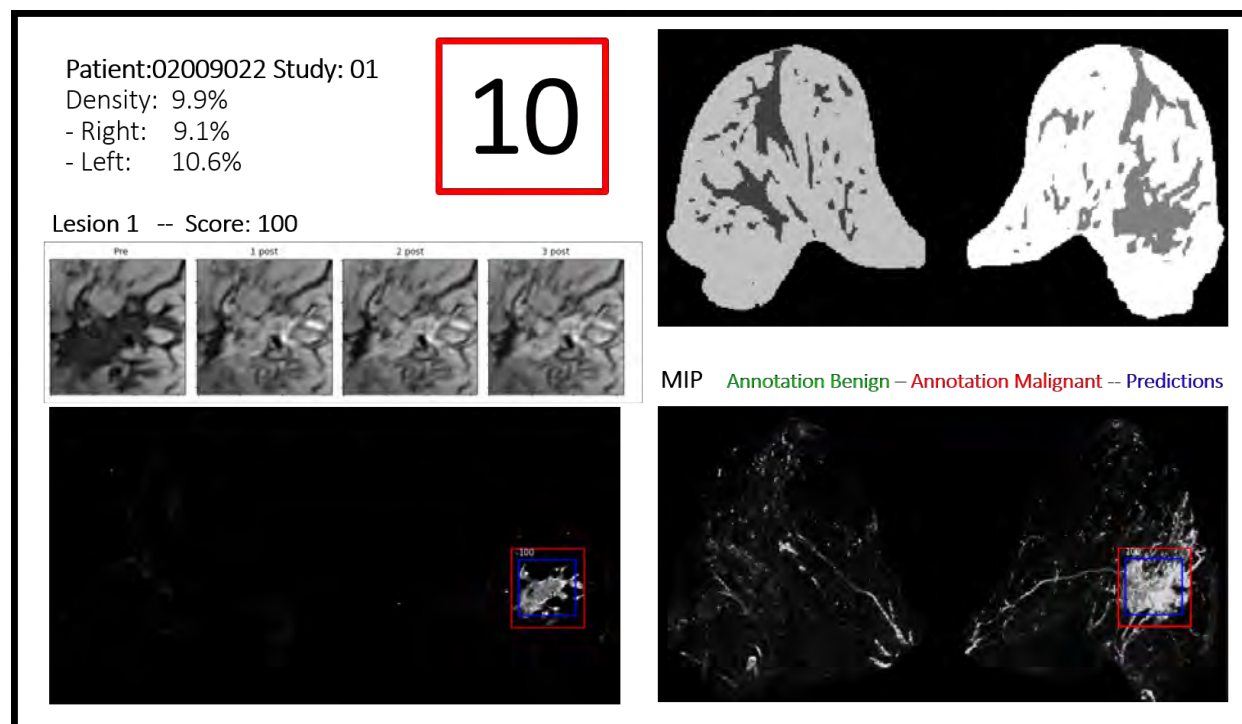
Figure 25: Overview of the predictions of our pipeline in a report sheet. The values of density, the segmentation mask, the global malignant score and all the detected lesions with corresponding score are shown. In this case, annotations of benign and malignant lesions are only displayed for illustrative purposes.

menting the lesion or lesions and generating new bounding boxes from the connected components, we mitigated this issue. This allows the network to be trained with more deterministic data. Moreover, Retina-Unet makes it possible to also use the extra information of the segmentation task by including an additional head. After a visual inspection of the predictions, it is evident that on several occasions the outputs of the network are more precise to each lesion than the original bounding boxes.

Training detection networks is usually harder than classification or segmentation ones, and indeed, this was the case. Although not being mentioned in the Retina-Unet paper, we found it convenient to train first from the segmentation branch and once we have meaningful features maps in the FPN, to add the back-propagation from the classification and regression heads. We expect this practice to be useful in other medical image domains.

Comparing between the model with patch size of 96 and 128, we observed that the one of 96 performs slightly better. This was not expected, as usually the larger patches benefit from the larger receptive field. To clarify, as we are only considering one training versus one training, the reason can be that the one with a larger patch size is stuck in a local minimum with lower performance than the one of 96. Also, because of the way that the sliding window was implemented by the authors, the small patch size benefits on average from a higher overlap. After Weighted Box Clustering

to gather the detections, can slightly boost the performance.

Trials with test time augmentations and multiple checkpoints do not improve performance and make it slower at inference time. Concerning the first one, similar augmentations are used for training, therefore, the network is robust to them and the results from the ensemble of augmentations do not vary significantly. We expect an analogous effect from the multiple checkpoints, as they are coming from the same model and similar in nature, therefore, the ensemble does not provide a major benefit. An ensemble between more diverse models can be considered as future work.

The score threshold of 0.25 is a hyperparameter of the pipeline. This value was used for training the classification, but it is not necessary to be the same at inference. Although we are using the same value, we expect that lowering the threshold can slightly improve performance, but at a cost of higher computational requirements. This is because the lower the score threshold, the higher the number of candidates that should enter the classification approach.

5.3. Classification

Our first experiment of Figure 17 compares the marginal increase in the classification performance when an extra post-contrast is included and the effect of using registration. It is clear the importance of the contrast agent injection when we analyze the additional

value of the first post-contrast. Also, including the first three post-contrasts has a significantly better performance than only using one (see 4.3.3). We decided not to concatenate the fourth post-contrast as in general DCE-MRI has at least three post-contrast but there is no guarantee that it has more. In terms of malignant versus benign categorization, the experiments showed that adding additional post contrasts to incorporate wash-out information does not improve classification ability significantly. It is evident, that a larger benign and malignant dataset would reduce the confidence intervals and reveal a more distinct trend. Although when using pre-contrast, first and the last post-contrast, the results look promising, we opted to exclude this line of work as the network may learn that normal patients have fewer post-contrasts on average, which would introduce a bias.

Concerning the use of registration, there is a consistently positive contribution on the AUC values. This implies that the registration parameters achieve the right balance in the trade-off between correcting motion artifacts and preventing lesion deformation. However, because it is a time-consuming operation as it should be performed for the three post-contrasts, we decided not to use it for our current pipeline.

Regarding the value of T2 and diffusion sequences, the experiments in Figure 18 show that there is not enough evidence to refute the null hypothesis that they modify the classification performance. As the normals do not have a T2 or diffusion volume, our dataset of malignants and benigns is small, therefore, drawing conclusions is difficult due to the huge uncertainty intervals. In addition, the difficulty of registering diffusion volumes limits the ability of a CNN to exploit this additional information, which does not occur with T2. Through Deep Learning, we might explore the possible benefits of T2w with enough data, as its value has been questioned by Mann et al. (2019a).

Occlusion sensitivity proved to provide valuable information about which sectors of the candidate patch are more relevant for the classification. By knowing that the lesion area affects the output probability the most, it is possible to rule out certain biases, such as different intensities distributions between normal and malignant images.

Besides the ResNet18, we implemented a Vision Transformer for classification. When the volumes are concatenated in the channel direction, the results are comparable to the convolutional architecture. However, the main goal was to use the extra flexibility of transformers to relate tokens in different positions by using only one channel and concatenating in the axial direction. The results show strong evidence that the extra flexibility is not beneficial, at least for the size of our dataset. The motion misalignment is small enough to be able to use non-registered volumes as channels, while a more flexible technique does not have a clear advantage. Also, the ViT tokens are large in comparison with the

motion artifacts. We expect the Vision Transformer to outperform a CNN for adding information of a diffusion image that is difficult to register. For future works, a Cross View Transformer (Tulder et al., 2021) that combines the benefits of CNNs and transformers can be considered.

For our final pipeline, we used the ResNet18 trained on pre-contrast and the non-registered first three post-contrasts. In general, every breast DCE-MRI contains these volumes, and not using registration makes it faster at inference time. In terms of performance, the AUROC is 0.964 (0.946-0.978) which value corresponds to those recently published by Witowski et al. (2022). The AUROC of malignant versus benign is 0.778 (0.683-0.853). This decrease in performance was expected as benign annotated were the ones that radiologist have doubts about being malignants, hence they are consider as difficult benign lesions. Apparently, it can be reason why there is no meaningful improvement with the addition of more than one post-contrast for these lesions.

5.4. Pipeline discussions

In terms of system performance, the classification step improves it by reducing the false positives (Figure 23). The final CPM value is 0.932 (0.890-0.964) outperforming the previous system by Dalmış et al. (2018) for lesion detection on a similar dataset which obtains a CPM score of 0.6429. In the field of ultra-fast DCE-MRI, a recent publication of a CADe based on a detection 3D RetinaNet achieves a CPM of 0.86 (Ayatollahi et al., 2021). Translating into predicted values, with a threshold of 5% for the score at a scan level, we achieve a NPV=0.941(0.892-0.986) at a PPV=0.634(0.530-0.721) considering benigns and normals as negatives.

This accomplishment is achieved because of the good performance in each stage. Segmentation dice scores correspond to state of the art values for our internal dataset, allowing for proper masking and cropping. For the detection stage, we believe a critical factor is the generation of new bounding boxes for training with less variability and the capacity of Retina-Unet of benefiting from the lesion segmentations. Finally, for the last stage, the ResNet18 is specifically designed for classification by being deeper than the classification head of the detection network and including residual connections. In addition, the input of the detection is the relative enhancement volume while for the classification is the pre-contrast and the first three post-contrast volumes, therefore, giving more information and flexibility to the network. Aside from this, there is a subtle difference worth mentioning. While the detection network uses patches around a malignant lesion as negative examples for training, the classification only relies on benign lesions or detections in normal studies for the negative class. In medical imaging, invasive cancerous lesions do not have clear boundaries, so training with

negative patches around a malignant lesion can be misleading. The possibility of using a different and better curated dataset for the classification step is a key factor in the performance.

5.5. Limitations and future work

A limitation is the homogeneous dataset used in this project, which does not allow generalization to other MRI protocols. Moreover, the wide confidence intervals limit the conclusions we can draw due to the small dataset. Future research should concentrate on training and validating similar pipelines on larger multi-protocol datasets in order to positively impact the clinical practice. Our method might potentially serve as a basis for fat-suppressed and ultra-fast protocols as well.

6. Conclusions

In this master thesis, an automated breast lesion detection CADe system for DCE-MRI was designed. Our final pipeline starts with the segmentation of FGT and fatty tissue using U-Net, achieving a dice score for the whole breast of 0.964 ± 0.018 on our internal dataset. From the segmentation masks, the percentages of FGT are calculated and reported as density estimations. A Retina-Unet was implemented for the detection of candidate lesions benefiting from the lesion segmentation signal, followed by a ResNet18 applied on each candidate for malignant classification. Our proposed pipeline achieved a CPM value of $0.932(0.89-0.964)$ outperforming the previous system for a similar dataset and could be the foundations of a system that supports radiologists in interpreting DCE-MRI studies.

References

Arponen, O., Masarwah, A., Sutela, A., Taina, M., Könönen, M., Sironen, R., Hakumäki, J., Vanninen, R., Sudah, M., 2016. Incidentally detected enhancing lesions found in breast MRI: analysis of apparent diffusion coefficient and T2 signal intensity significantly improves specificity. European radiology 26, 4361–4370.

Ayatollahi, F., Shokouhi, S.B., Mann, R.M., Teuwen, J., 2021. Automatic breast lesion detection in ultrafast DCE-MRI using deep learning. Medical physics 48, 5897–5907.

Baumgartner, M., Jäger, P.F., Isensee, F., Maier-Hein, K.H., 2021. nndetection: A self-configuring method for medical object detection, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 530–539.

Dalmış, M.U., Litjens, G., Holland, K., Setio, A., Mann, R., Karssemeijer, N., Gubern-Mérida, A., 2017. Using deep learning to segment breast and fibroglandular tissue in MRI volumes. Medical physics 44, 533–546.

Dalmış, M.U., Vreemann, S., Kooi, T., Mann, R.M., Karssemeijer, N., Gubern-Mérida, A., 2018. Fully automated detection of breast cancer in screening MRI using convolutional neural networks. Journal of Medical Imaging 5, 014502.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

Gubern-Merida, A., Kallenberg, M., Mann, R.M., Marti, R., Karssemeijer, N., 2014. Breast segmentation and density estimation in breast MRI: a fully automatic framework. IEEE journal of biomedical and health informatics 19, 349–357.

Gubern-Mérida, A., Martí, R., Melendez, J., Hauth, J.L., Mann, R.M., Karssemeijer, N., Platel, B., 2015. Automated localization of breast cancer in DCE-MRI. Medical image analysis 20, 265–274.

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H.R., Xu, D., 2022. Unetr: Transformers for 3d medical image segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574–584.

Huo, L., Hu, X., Xiao, Q., Gu, Y., Chu, X., Jiang, L., 2021. Segmentation of whole breast and fibroglandular tissue using nnU-Net in dynamic contrast enhanced MR images. Magnetic Resonance Imaging 82, 31–41.

Jaeger, P.F., Kohl, S.A., Bickelhaupt, S., Isensee, F., Kuder, T.A., Schlemmer, H.P., Maier-Hein, K.H., 2020. Retina U-Net: Embarrassingly simple exploitation of segmentation supervision for medical object detection, in: Machine Learning for Health Workshop, PMLR. pp. 171–183.

Jiao, H., Jiang, X., Pang, Z., Lin, X., Huang, Y., Li, L., 2020. Deep convolutional neural networks-based automatic breast segmentation and mass detection in DCE-MRI. Computational and Mathematical Methods in Medicine 2020.

Klein, S., Staring, M., Murphy, K., Viergever, M.A., Pluim, J.P., 2009. Elastix: a toolbox for intensity-based medical image registration. IEEE transactions on medical imaging 29, 196–205.

Lehman, C.D., Schnall, M.D., 2005. Imaging in breast cancer: magnetic resonance imaging. Breast Cancer Research 7, 1–5.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Mann, R.M., Cho, N., Moy, L., 2019a. Breast MRI: state of the art. Radiology 292, 520–536.

Mann, R.M., Kuhl, C.K., Moy, L., 2019b. Contrast-enhanced MRI for breast cancer screening. Journal of Magnetic Resonance Imaging 50, 377–390.

Nassif, A.B., Talib, M.A., Nasir, Q., Afadar, Y., Elgendy, O., 2022. Breast cancer detection using artificial intelligence techniques: A systematic literature review. Artificial Intelligence in Medicine , 102276.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics 9, 62–66.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Samperna, R., Moriakov, N., Karssemeijer, N., Teuwen, J., Mann, R., 2022. Annotation efficient breast and fibroglandular tissue segmentation using nnUNet in breast MRI .

Sung, H., Ferlay, J., Siegel, R.L., Laversanne, M., Soerjomataram, I., Jemal, A., Bray, F., 2021. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 can-

cers in 185 countries. CA: a cancer journal for clinicians 71, 209–249.

Tulder, G.v., Tong, Y., Marchiori, E., 2021. Multi-view analysis of unregistered medical images using cross-view transformers, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 104–113.

Vidal, J., Vilanova, J.C., Martí, R., et al., 2022. A U-Net Ensemble for breast lesion segmentation in DCE MRI. Computers in Biology and Medicine 140, 105093.

Wilcoxon, F., 1992. Individual comparisons by ranking methods, in: Breakthroughs in statistics. Springer, pp. 196–202.

Witowski, J., Heacock, L., Reig, B., Kang, S.K., Lewin, A., Pysarenko, K., Patel, S., Samreen, N., Rudnicki, W., Łuczyńska, E., et al., 2022. Improving breast cancer diagnostics with artificial intelligence for MRI. medRxiv .

Yamaguchi, K., Schacht, D., Newstead, G.M., Bradbury, A.R., Verp, M.S., Olopade, O.I., Abe, H., 2013. Breast cancer detected on an incident (second or subsequent) round of screening MRI: MRI features of false-negative cases. American Journal of Roentgenology 201, 1155–1163.

Zeiler, M.D., Fergus, R., 2014. Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer. pp. 818–833.

Zhang, Y., Chan, S., Park, V.Y., Chang, K.T., Mehta, S., Kim, M.J., Combs, F.J., Chang, P., Chow, D., Parajuli, R., et al., 2020. Automatic detection and segmentation of breast cancer on MRI using mask R-CNN trained on non–fat-sat images and tested on fat-sat images. Academic Radiology .
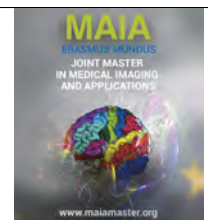
Deep convolutional neural networks for the analysis of retinal damage in optical coherence tomography images

Anastasiia Rozhyna, Henning Müller, Manfredo Atzori

Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO Valais) 3960-Sierre, Switzerland

Abstract

Background: Retinal damage is the ultimate cause of vision loss. **Objective:** This thesis aims at investigating retina diseases by analyzing the OCT retina images, targeting the development of a multi-purpose diagnostic tool.

Methods: This research proposes a transfer learning approach with continual learning to perform classification in retinal OCT scans. The first part of the work presents an overview of datasets for OCT images and the current challenges in ophthalmology. Second, several deep neural network architectures are evaluated for classification purposes using the original classes provided in the datasets. Third, it is investigated the possibility to study relationships between retina alterations and diseases of the central nervous system (for instance, multiple sclerosis), evaluating if it is possible to take advantage of applying previous knowledge and abilities to novel tasks.

In this work, we use pre-trained CNN architectures such as VGG16, VGG19, ResNet50, MobileNet, InceptionV3 and Xception with the pre-trained weights on the ImageNet dataset to reduce the training time and increase the performance. The proposed approach was evaluated on two different datasets. The Large Labeled Optical Coherence Tomography (OCT) Images (LLOCT dataset) and The Multiple Sclerosis and Healthy Controls dataset (MS and HC dataset) were used for training and testing the approach for OCT classification applying transfer learning.

Results: The results indicate that the proposed method of transfer learning is a very promising tool for classifying multi-class retinal OCT scans. The obtained results demonstrate the best performance on LLOCT dataset which was achieved by VGG16 with classification accuracy of 96 %.

Keywords: Optical Coherence Tomography, OCT, Retina Images, Deep Convolutional Neural Networks, Transfer Learning

1. Introduction

In recent years, the number of cases of eye diseases has increased strongly due to many factors such as diet, lifestyle, increased life span and also genetics. Today's statistics show that the number of people with eye problems is set to increase each year in the next decade. Overall, around 2.2 billion people have eye conditions and vision problems. Among them, at least 1 billion people suffer from a vision problem that could have been avoided. Therefore, there is an urgent need for practical, high-quality interventions and fast methods for diagnosing eye diseases. Eye disorders that can cause vision impairment and blindness are at the front line of prevention and intervention initiatives (Organization, 2019).

The WHO World report on the vision from 2016 to 2030 demonstrates that the number of people with eye conditions will increase in the following years due to different causes. By 2030 the estimated number of people suffering from diabetic retinopathy, glaucoma and age-related macular degeneration around the world will reach 95.4 and 243.4 million, respectively. Globally, the predicted number of persons living with Multiple Sclerosis (MS) has risen to 2.8 million by 2020. The estimate is 30% higher than in 2013, when using the same methods as in 2013. Based on current statistics, health systems face enormous problems in satisfying current eye-care demands and the situation is expected get even worse in the future. Ophthalmological imaging is a promising and helpful tool in modern ophthalmol-
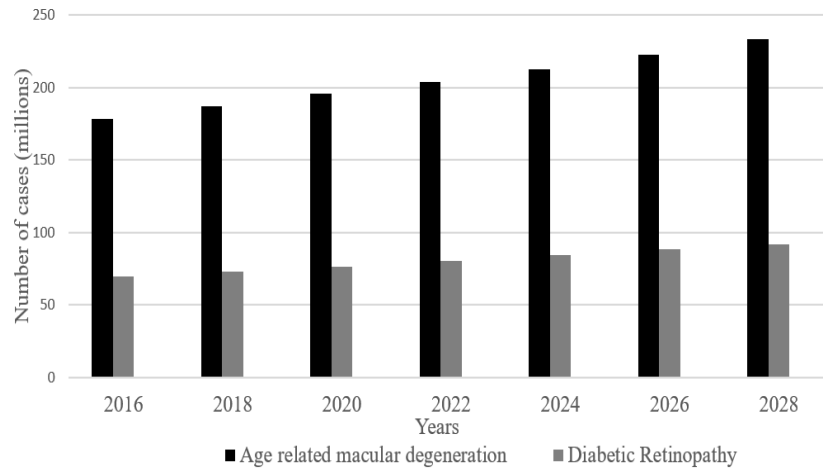
Figure 1: Projected number of people worldwide with age-related macular degeneration and diabetic retinopathy (up to the year 2028, World Health Report on Vision)

ogy. In recent years eye healthcare systems advanced quickly by applying deep learning algorithms for better understanding of eye imaging (Litjens et al., 2017). Retinal screening for all patients is crucial in today's health care systems for the early detection of eye diseases. Most diseases are asymptomatic in the early stages, which motivated the search for rapid diagnosis in screening programs, that can help to stop progression and avoid vision loss if these diseases are detected at an early stage. There are currently many undiagnosed and untreated cases of vision pathologies. The demand for automated analysis for identifying eye images has increased not only due to the scarcity of ophthalmologists but also to increase the accuracy and efficiency of the in-time diagnosis. The development of computerized tools for the analysis of OCT images is a key step in providing ophthalmologists with a complete examination.

In this study, we aim to classify various pathologies such as Choroidal neovascularization (CNV), Diabetic macular edema (DME), and Multiple drusen present in early age-related macular degeneration (AMD) as well as Multiple Sclerosis (MS) on OCT images. Each of these pathologies can be detected on retinal OCT scans. The main idea of this work is to use existing transfer learning methods and apply this approach to the most widely spread vision diseases that are seen on OCT images. When making a comparison between two modalities, retinal fundus images and OCT, both have advantages and disadvantages. Fundus imaging is a simple, inexpensive and quick procedure that can be performed with usual cameras. Previously, fundus imaging was the most frequently used diagnostic modality for detecting retinal disorders.

However, OCT imaging is now frequently used for detecting retinal conditions because of its capacity to detect even minor changes in the retinal layers (Arabi

et al., 2017). In terms of eye diseases, it is much easier to use OCT scanning for the detection of conditions that influence the layer thicknesses and structure of retina.

1.1. Retinal OCT & Diseases

Optical coherence tomography (OCT) is a non-invasive diagnostic imaging method that uses optical characteristics to reconstruct cross-sections of tissues. It is commonly used in ophthalmology to image the anterior eye and retina structure for diagnostic purposes. OCT gives histological details and can be called an optical biopsy. The main advantages of OCT are the fast procedure, non-invasiveness and its reproducibility.

The basic principle of OCT is the estimation of the tissue depth. OCT has many applications but retinal imaging is one of the most common and important uses of OCT. Retinal imaging is employed for the detection and diagnosis of retinal diseases. Many serious diseases can be present in the retina and have their origins in the eye or brain. The most common diseases can be studied using eye imaging and image processing. OCT is similar to ultrasound imaging, except that it detects reflections of near infrared light instead of sound. It uses infrared light to provide a high-resolution 3D view of living tissues with a depth of a few hundred microns. It generates 2D and 3D images using low coherence interferometry (Eladawi et al., 2018).

Hundreds of eye disorders and vision impairments exist. The one in the following list are among the most frequent eye conditions that result in vision loss or blindness.

Diabetes mellitus is diagnosed if a patient has a fasting plasma glucose of over 7.0 mmol/l according to the World Health Organization (WHO). The causes of this disease can be very different from genetics to a sedentary lifestyle. Treatment is a diet change and strict in-

sulin control. The progressive state of this disorder can result in diabetic retinopathy, which is a retinal complication of diabetes.

Diabetic Retinopathy (DR) is the leading cause of complete or partial blindness among people with diabetes mellitus. DR can be diagnosed early with the identification of retinal lesions on the surface of the eye. In the eye, hyperglycemia damages the retinal vessel walls, which can lead to ischemia or breakdown of the blood-retinal barrier.

Age-related macular degeneration (AMD) is the most common cause of visual loss. It has become a rapidly growing public health problem over the past years. Possible treatments include dietary supplements that can help to slow down the disease.

Glaucoma is the third leading cause of blindness. This disorder can be characterized by gradual damage to the optic nerve and resultant visual field loss. Early diagnosis and optimal treatment can minimize the risk of visual loss.

Cardiovascular diseases can be seen in the retina in different ways. Hypertension and atherosclerosis cause changes in the ratio between the diameter of retinal arteries and veins. Even direct retinal ischemia can occur because of hypertension, leading to visible retina damage spots.

Central nervous system diseases (CNS) including Alzheimer's disease (AD), Multiple sclerosis (MS) and cerebrovascular disorders were shown to have pathological changes in the retina. Ophthalmic examinations can analyze the retina and the optic nerve of the central nervous system (Landau and Kurz-Levin, 2011). Various studies have shown that neurological illnesses may cause changes in the retina. Several investigations, for example, have shown that thinning of retinal layers occurs in Alzheimer's disease. Still, these studies are limited by the lack of a consistent imaging procedure that would allow for more quantitative analyses. Furthermore, the mechanisms with retinal changes in neurological illnesses are still not fully understood.

2. State of the art

Convolutional Neural Networks (CNNs) are a type of artificial neural network that evolved from standard artificial neural networks and has showed promise in image classification, object recognition and image segmentation. In ophthalmology, CNN has recently been used to detect diabetic retinopathy and macular fluid in fundus images. CNNs should be able to extract information for retinal classification from OCT scans. A convolutional neural network uses a local connection and weight sharing strategy, which simplifies the network's parameters and model complexity, making the deep network easier to modify. CNNs combine approaches for feature extraction and feature classification in an end-to-end approach.

The CNN is an end-to-end feature learning method that can automatically learn a hierarchy of features from a given training sample, as opposed to the classical manual (handcrafted) feature extraction processes that often use a classifier in a second step.

Triwijoyo et al. (2017) worked with retinal images to diagnose disorders like glaucoma and hypertension in the eyes. They claim that recognizing vascular irregularities in retinal images can help physicians in diagnosing and treating stroke, cerebral injury, artery disease, and other conditions early. To recognize retinal images, they used Convolutional Neural Networks (CNN) as a classifier. They used the STARE fundus color image dataset and divided into 15 categories. The CNN model was shown to have an accuracy of 80.93 percent in the experiments.(Eladawi et al., 2018)

Luo et al. (2021) proposed a semi-supervised deep learning method built upon pre-trained VGG-16 and virtual adversarial training (VAT) for the detection of retinopathy for automatically diagnosing retinopathy using only 80 OCT images from the Large Dataset of Labeled Optical Coherence Tomography (OCT) Images dataset. As a result the proposed technique achieves classification accuracies of 0.942 and 0.936, sensitivities of 0.942 and 0.936, specificities of 0.971 and 0.979, and AUCs of 0.997 and 0.993, respectively.

Zhang et al. (2019) analyzed OCT scans using a feature extractor based on a pre-trained ResNet50 from the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) and support vector regression. Using a data set of 482 OCT images, they revised and fine-tuned the network, achieving an accuracy of 0.93.

To identify retinal OCT images, the suggested classification approach by Li et al. (2019) used an ensemble of four classification model instances, each of which was based on a residual neural network (ResNet50). On a developing retinal OCT imaging dataset, the experiment used a patient-level 10-fold cross-validation approach. At the B-scan level, the proposed approach achieved a classification accuracy of 0.973 %, a sensitivity of 0.963, and a specificity of 0.985, matching or exceeding that of ophthalmologists with significant clinical experience.

Karri et al. (2017) demonstrate an approach for identifying retinal pathologies on OCT images. Their method fine-tunes a pre-trained convolutional neural network (CNN), GoogLeNet, to increase its prediction capabilities and finds responses during prediction to better comprehend learned filter features.Their model achieved 0.94% classification accuracy for diabetic macular edema and dry age-related macular degeneration.

Wang and Wang (2019) use two separate sources of OCT datasets to provide an automated method based on deep learning to classify DME and AMD.The approach consisted of using CliqueNet, DPN, DenseNet, ResNet, ResNext neural network on two public OCT datasets.

On the public OCT dataset 1, the AUC values obtained by DenseNet, ResNet, DPN, ResNext, and CliqueNet are all over 0.96, while the average AUC values of three types are all above 0.97.

Fang et al. (2019) proposed a new lesion-aware convolutional neural network (LACNN) method for retinal OCT image classification, in which the CNN is guided by retinal lesions inside OCT images to obtain a more accurate classification.

Naz et al. (2017) worked on identifying DME by automatically classifying optical coherence tomography (OCT) pictures. They proposed a realistic and very easy method for robust DME classification based on OCT image information and coherent tensors. The features retrieved from thickness profiles and cysts were tested using the Duke Dataset, which included 55 sick and 53 normal OCT scans. The support vector machine with leave-one-out has the maximum accuracy of 79.65 percent, according to the comparisons.

On the basis of CNNs, Rong et al. (2019) proposed a surrogate-assisted retinal OCT picture classification approach. To analyze the performance of the proposed technique at the B-scan level, two databases were used: a local database and a public database Duke. The results demonstrate that the proposed method is a very promising tool for automatically classifying OCT pictures (AUC of 0.9783 in the first dataset and AUC of 0.9856 in the second dataset).

Khan et al. (2022) proposed a continuous learning that allows deep learning models to effectively store prior information while adapting to new classes, datasets and applications. The approach included 9 publicly available and multimodal datasets for three applications, namely learning classification of items from X-ray baggage scans, learning to predict pneumonia from chest X-ray scans and finally classification of retinal diseases from multimodal imagery (Fundus and OCT modalities). The proposed framework included class continual learning as well as dataset continual learning in the cross-domain learning. The top-1 achieved accuracy is 0.9863% and F-1 score of 0.993.

A deep multilayered CNN for eye illness detection and classification proved to be good approach with high classification accuracy. The previous methods focus on typical classes of diseases, not taking into account neurological diseases and their classification. We would like to take advantage of this opportunity, as compared to past works and to test transfer learning for typical eye disease problems like diabetic eye disease, age related macular degeneration etc. and for multiple sclerosis classification in retinal OCT scans. In this way, we will be able to test how strongly biomarkers of eye disease are related between each other and if it is possible to have one tool for classification and detection of various pathologies. Starting from the most recent works in literature, the proposed approach is to apply the transfer learning for the classification. As a result of the previous investigations, we are considering a method to this problem for the future work by employing methodologies from the field of continuous learning to acquire knowledge across multiple tasks without retraining. Researchers have used incremental learning to adjust deep neural networks to learn multiple classification tasks with a limited number of training data. By combining this strategy with an incremental learning framework, classification performance can be expected to improve.

3. Materials

3.1. Datasets Overview

Large datasets are a vital part for training deep neural networks, and thus to allow speeding up research based on health data too. Table 1 presents datasets with OCT as main imaging modality (Khan et al., 2021). The total number of publicly available datasets is 17. Of the 17 datasets, we found 6 datasets from USA and 1 dataset in collaboration between USA and China, 4 datasets from Iran, 2 datasets from Spain, 1 dataset from India and 3 datasets with unknown origin country. Ophthalmological diseases that are represented in the datasets include diverse eye conditions, such as age-related macular degeneration,diabetic eye disease, glaucoma etc. Some datasets include samples of healthy eyes.

Among the 17 ophthalmological imaging datasets based on OCT modality, part of the datasets contained 2 dimensional imaging data and the other part contained 3 dimensional imaging data. Most datasets stored images in MAT, TIFF or JPEG formats (Khan et al., 2021). The diseases representation is uneven. Diabetic retinopathy, and age-related macular degeneration were disproportionately overrepresented in contrast to other eye illnesses.

3.2. Large Dataset of Labeled Optical Coherence Tomography (OCT) Images

The first dataset used in this work is the Large Dataset of Labeled Optical Coherence Tomography (OCT) Images obtained from the University of California San Diego in collaboration with Guangzhou Women and Children's Medical Center (Kermany et al., 2018a).

Thousands of validated OCT and Chest X-Ray pictures are included in this dataset, only part of OCT scans was used in this work. which was reported and analyzed in "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning"(Kermany et al., 2018b). Image resolution is 512x496 pixel size. The images are fully anonymized and do not contain any personal patient data. The format of the images is JPEG. The access to the dataset is public available. The dataset includes 4 different conditions: Choroidal Neovascularization (CNV),Diabetic Macular Edema (DME), Age-related macular degeneration (Drusen), Healthy eyes (Normal).

Dataset name	Country	Number of of patients	Number of images	Eye disease	File format
2014 Srinivasan	USA	45	3231	Diabetic eye disease age-related macular degeneration healthy eyes	TIFF
Contact Lens Anterior Segment-OCT Understanding Dataset	Spain	16	112	NR	JPEG
Corneal Heidelberg OCT	Iran	15	579	Healthy eyes	MAT
Retinal Fundus and OCT	Iran	22	44	Various retinal diseases diseases	MAT and JPEG
2012 Fang	USA	17	51	Age-related macular degeneration and healthy eyes	TIFF
Duke OCT	USA	384	38400	Age-related macular degeneration and healthy eyes	MAT
Kermany/Guangzhou	USA China	5319	109312	Diabetic eye disease, choroidal neovascularization drusen, healthy eyes	JPEG
Noor Hospital	Iran	148	4142	Diabetic eye disease age related macular degeneration healthy eyes	TIFF
2015 Chiu	USA	10	10	Diabetic eye disease	MAT
Healthy OCT and Fundus	NR	50	100	Healthy eyes	MAT and JPEG
OCT Glaucoma Detection	NR	624	1100	Glaucoma and healthy eyes	NumPy Array File
OCTAGON	Spain	213	213	Diabetic eye disease and healthy eyes	JPEG and TIFF
OCT Retinal Image Analysis 3D	NR	10	10	Healthy eyes	MAT
Canada OCT Retinal Images	India	NR	470	Diabetic eye disease, healthy eyes age-related macular degeneration macular hole central serous retinopathy	JPEG
Retinal OCT Classification Challenge	Iran	NR	165	Diabetic eye disease, healthy eyes	MAT
2011 Chiu	USA	20	220	Age-related macular degeneration	MAT
OCT MS and Healthy Controls Data	USA	35	1715	Multiple sclerosis healthy eyes	VOL

Table 1: Publicly available OCT imaging datasets

Class	Train	Test
CNV	37205	250
DME	11348	250
DRUSEN	8616	250
NORMAL	51140	250

Table 2: Data distribution in Large Labeled Optical Coherence Tomography (OCT) Images dataset

Kermany et al. (2018a) divided the OCT images into a training set and a testing set. The training set included 108,309 images, comprising 37,205 images of CNV, 11,348 images of DME, 8,616 images of Age-related macular degeneration (Drusen), and 51140 images of a healthy eye conditions. The scan resolution varied slightly between subjects, the resolution has a 5.7 µm pixel size in LLOCT dataset.

The data was received from big number of patients for each pathology, for instance with Diabetic Macular Edema (DME) the total number of testing patients is 709, for Choroidal Neovascularization is 791 patients, for Drusen is 713 number of patients and the number of healthy cases is 3548. Patient characteristics such as

number of patients, age and gender are included in Table 3. It demonstrates the characteristics of patients whose OCT scans were included in the analysis. This table represents well in detail the dependence of a particular disease on age and gender. For instance, in cases with Age-related macular degeneration which is associated with the Drusen class the average age of patients is 82 years old.

3.3. OCT Multiple Sclerosis and Healthy Controls Dataset

As can be seen from the analysis of datasets, the distribution of diseases is quite uneven, and the representation of several eye diseases is relatively small. Therefore, there is a special interest in working with datasets that have non-typical classes of diseases.

In patients with multiple sclerosis, a thinner layer of the retina and others unique biomarkers of the disease are observed. For example, patients with advanced multiple sclerosis have been found to have thinner layers of retinal nerve fibers and reduced macular volume (Petzold et al., 2010).

The OCT Multiple Sclerosis and Healthy Controls Dataset was used for analyzing the special connection between the central nervous system and specific retinal damage (He et al., 2019a).

This dataset was obtained from Johns Hopkins School of Engineering (He et al., 2019d). The dataset contains 35 OCT retinal scans using a Spectralis OCT system (Heidelberg Engineering, Heidelberg, Germany), 21 of the 35 participants had been diagnosed with Multiple Sclerosis (MS), whereas the other 14 were Healthy Controls (HC) (He et al., 2020). The scan resolution varied slightly between subjects as well as in the first dataset.The MSHC dataset has a a mean over all the subjects of 5.8 μm. The automatic real-time function is used to acquire the scans from the Spectralis scanner.

The volume data was exported from the Spectralis scanner using the.vol file format. Each volumetric OCT image file has 49 scans. Each scan has a total size of 496x1024. The dataset divided as following the training set consists of 929 scans for Multiple Sclerosis and 586 for Healthy Controls. For the testing set both classes have 100 scans each. All scans were checked for microcystic macular edema (a pathology sometimes found in MS subjects). The participants' ages were between of 20 and 56, with an average age of 39 and received data was captured from the right eyes(He et al., 2019c). In this dataset, scans were extracted from a 3D volume files. For the purpose of preparing the dataset and establishing format uniformity, it was decided to extract images from the volume format and convert them into JPEG format. The process was done in Matlab with provided scripts allowing to read from raw Spectralis (.vol) (He et al., 2019b).

4. Methodology

4.1. Data Pre-processing

The classification performance of retinal OCT scans is affected by artifacts. To remove the artifacts, binary transformation was applied to the retinal OCT images using an experimental threshold value. There were several stages in the data pre-processing algorithm. Figure 4 demonstrates a flowchart of data pre-processing.

First, simply apply a binary transformation to the input image. Apply a bounding box across the dark pixels and crop the region inside after converting white pixels to complete dark pixels. Finally, resize the image to 150x150 pixels (Sezgin and Sankur, 2004). Each OCT scan was processed as a 150x150x3 image, where 3 is the amount of color channels, to retain compatibility with the CNN-based architecture. Resizing images is an important pre-processing step. Models are mostly trained faster on smaller images. Many deep learning model architectures demand that images have the same size, despite the fact that raw images may differ. All images must be resized to a fixed size before being fed into the CNN. This process helps with less deformation of the image's features and patterns during the training process.

4.2. Image augmentation

The processed image has been resized for use in the deep CNN architecture.

Deep convolutional neural networks require a large amount of training data to learn the data representation and perform effectively without overfitting.

Image augmentation is a popular technique for creating a powerful model that can be trained with a small number of training data. Image augmentation employs a variety of alteration techniques such as random rotation, flipping, image resizing, and a variety of other augmentation techniques (Yang et al., 2022). Because biomarkers and lesions can appear in a variety of orientations, data augmentation can be useful in OCT scans, leading to many copies of training samples.

By developing synthetic samples, data augmentation aims to improve the sufficiency and diversity of training data. The augmented data can be viewed as coming from a distribution that is similar to the genuine one. The following strategies are used for data augmentation:

- Horizontal Flipping is far more common than flipping the vertical axis. This augmentation is one of the simplest to employ and has shown to be effective on ImageNet datasets (Shorten and Khoshgoftaar, 2019).

- Rotation is in between 1° and 359° for rotation augmentations. The rotation degree parameter has a significant impact on the safety of rotation augmentations. In this work, slight rotation (10) is performed.
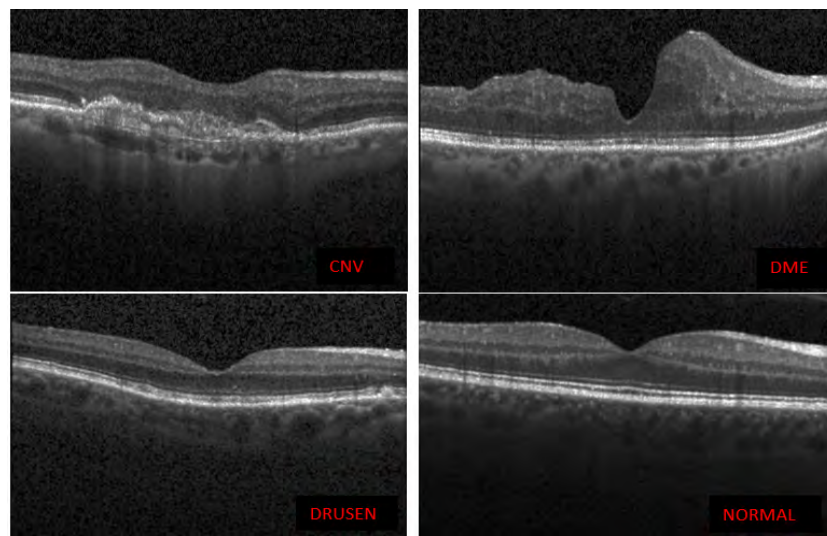
Figure 2: Samples from Large Dataset of Labeled Optical Coherence Tomography (OCT) Dataset. Panels present images: upper left – choroidal neovascularization (CNV); upper right – diabetic macular edema (DME); down left– drusen; down right – normal
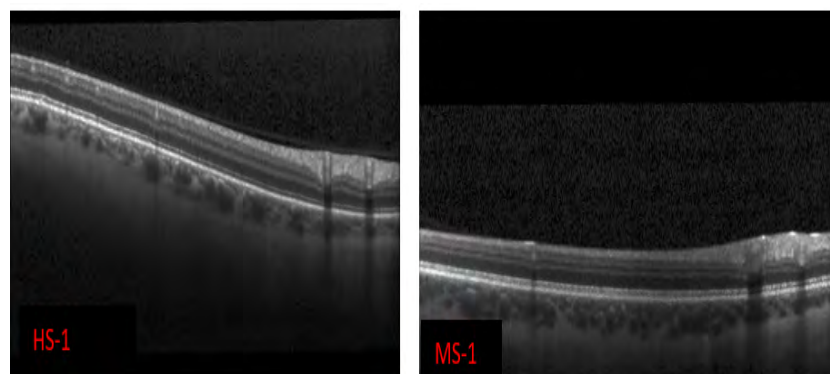
Figure 3: The data samples from Multiple Sclerosis and Healthy Controls Dataset. Panels present images: left image – healthy control sample, right image – multiple sclerosis sample
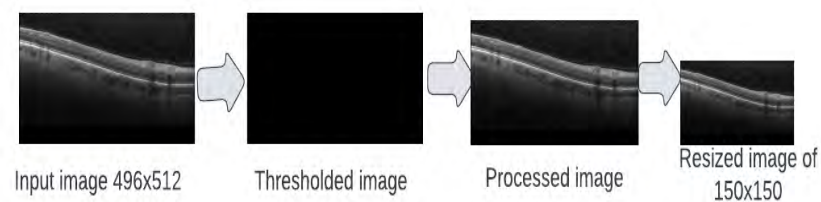
Figure 4: Data Preprocessing

16.7

Diagnosis	Diabetic Macular Edema (DME)	Choroidal Neovascularization (CNV)	Drusen	Normal
Number of Patients	709	791	713	3548
Mean Age (years)	57 (Range: 20-90)	83 (Range: 58-97)	82 (Range: 40-95)	60 (Range: 21-86)
Gender				
Male	38.3%	54.2%	44.4%	59.2%
Female	61.7%	45.8%	55.6%	40.8%

Table 3: The additional dataset information with patients analysis
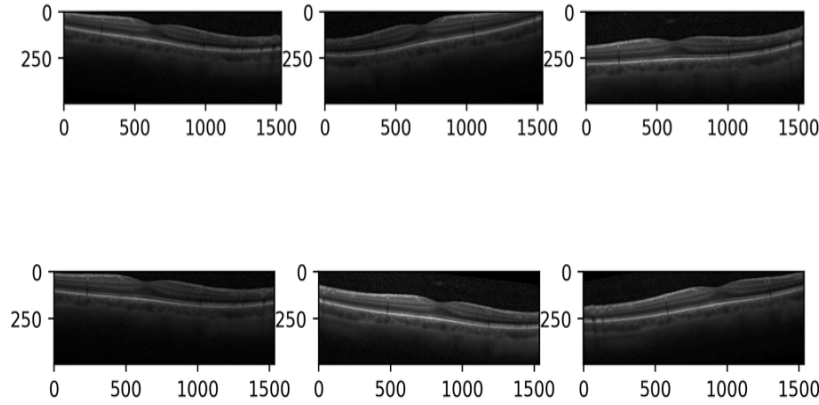
Figure 5: Data augmentation

Class	Train	Test
Multiple Sclerosis (MS)	929	100
Healthy Controls (HC)	586	100

Table 4: Data distribution in Multiple Sclerosis and Healthy Controls Dataset

- Rescaling (1./255) is to transform every pixel value from range [0,255] to [0,1]. The advantages include treating all photos the same way and employing a standard learning rate. Some images have a wide pixel range, while others have a narrow pixel range. All of the photos have identical model, weights, and learning rate. The image with an extensive range produces a stronger loss, whereas the image with a low range produces a lesser loss.

- Zoom is by using a value to specify the zoom-in value. The zoom was given as [0.7, 1] during data augmentation, which means 70 percent zoom in and 100 percent zoom out.

- Brightness is used with the goal is for a model to generalize across images with varying lighting levels. The brightness level set as (0.55-0.9).

- Shifting can help with better localization the object on the image. During shifting the value for width and height shifting set as (0.1)

- Fill Mode is used after image rotation. Because some pixels can migrate outside the image as it is rotated, leaving an empty region that must be filled in. In this case fill mode is used and can include a constant value, nearest pixel values, and so on. The default value for the fill mode option is "nearest" which simply replaces the empty region with the closest pixel values. Th fill mode was set as "nearest".

4.3. Proposed CNN Architectures

For our approach VGG16, VGG19, ResNet50,MobileNet, InceptionV3 and Xception were among the base model architectures used. They have already been trained on the ImageNet database.

VGG16 model was implemented in this work and was trained on Imagenet data containing 1.2 million color images and 1000 classes. All kernel sizes are 3x3 in the original VGG16, including 16 convolution layers using the Relu activation function. A max-pooling layer with all 2x2 kernel sizes follows each convolution layer. Convolution layers are used to retain training weights and act as an automatic feature extraction system. The final layer as a classifier is made up of three fully connected layers (FC). The weight of the training results can be stored by the convolution layer and FC, allowing them to determine the number of parameters(Simonyan and Zisserman, 2014a).

VGG19 is a convolutional neural network with 19 layers, 16 convolution layers, and 3 fully connected

layers for classifying images into 1000 object categories. The ImageNet collection, which comprises a million images in 1000 categories, was used to train VGG19. Due to the employment of numerous 3x3 filters in each convolutional layer, it is a particularly common method for image classification (Simonyan and Zisserman, 2014b).

ResNet50 is a pre-trained residual neural network model that is quite useful. The depth of ResNet is determined by the number of successive modules employed. Increasing the network's depth to achieve better precision, on the other hand, makes it more difficult to optimize the network.The input layer's residuals are learned by the network. Each block is made up of a succession of layers and a link that connects the block's input to its output. ResNet50 is made up of three successive convolutions, a 1×1, a 3×3, and a 1×1, as well as a connection that links the first convolution's input to the third convolution's output. ResNet50 model have a total of 25,636,712 parameters (He et al., 2016).

MobileNet model is built on depthwise separable convolutions, a type of factorized convolution that divides a standard convolution into a depthwise convolution and a pointwise convolution. The depthwise convolution's outputs are then combined using an 11 convolution by the pointwise convolution. A conventional convolution filters and combines inputs to create a new set of outputs in one step. This is separated into two layers by the depthwise separable convolution, one for filtering and the other for combining. MobileNet is a lightweight network that uses depthwise separable convolution to deepen the network and minimize parameters and computation compared to the VGG-16 network (Howard et al., 2017).

InceptionV3 is a 42-layer deep neural network with convolutions, max-pooling layers, average pooling, dropouts, and fully linked layers are among the symmetric and asymmetric building components in the Inception-v3 model. It's a widely used image recognition model that can achieve more than 78.1 percent accuracy on the ImageNet dataset (Szegedy et al., 2015).

Xception model adopted in this research is a pretrained ImageNet model provided on Keras that outperforms Inception V3 by a small percentage. The Xception architecture is a depthwise separable convolution layer stack with residual connections that are linearly stacked. The input format for the Xception is a 299x299 RGB image. It has a depth of 126 layers, with 36 convolutional layers for feature extraction. To reduce the number of parameters, a global average pooling layer is utilized to replace the fully-connected layer, and the softmax function is employed to output the prediction (Chollet, 2017).

4.4. Training Methodology

Transfer learning is typically applied in two ways: the first way is using a pre-trained model and replacing its last layers with others so that they can learn from the new data set.

Figure 6: The methodology

CNNs are commonly used for image categorization because of their strong performance for learning meaningful representations of images.

Our strategy is to use six pre-trained models as it was mentioned before. The ImageNet dataset was used to pretrain all of our networks. Specifically, instead of randomly initializing the parameters, we used parameters learned from the ImageNet dataset to initialize the parameters of our networks. Our networks were then fine-tuned to better fit the datasets using their pretrained parameters.

In Figure 8, you can see the suggested deep multilayered CNN. 13 convolutional layers, 2 fully connected layers, and 1 SoftMax classifier make up the VGG-16 model architecture. The deep multilayered CNN architecture was fed the processed images as input. Our input OCT image has a shape as 150x150x3 image, where 3 is the amount of color channels, to retain compatibility with the CNN-based architecture.

It passes through 2D convolutional layer, the output is 64 channels and after it another 2D convolutional layer.A kernel matrix is passed over the input matrix in the convolutional layer to build a feature map for the next layer.At this point we have 36928 parameters. The feature map output of a convolutional layer has the disadvantage of recording the exact position of features in the input.

This means that any tiny adjustments to the input image, such as cropping or rotation, will result in a completely new feature map. To address this issue, we use convolutional layer down sampling. A pooling layer can
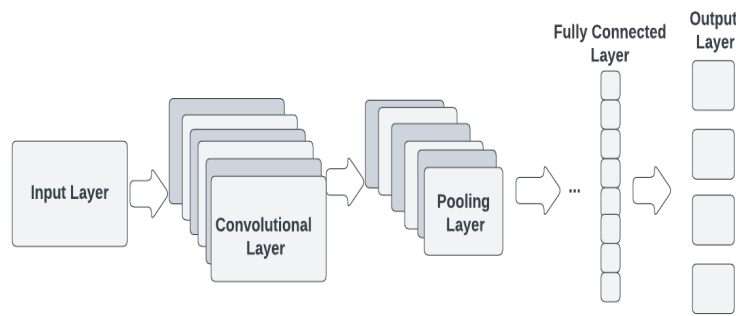
Figure 7: CNN architecture

be used to achieve down sampling. In order to reduce the size of feature maps, we 2D maxpooling and have a shape of 75x75x64. After it passes through two 2D convolutional layers again and double the size of filter from 64 to 128. The parameter number is 147,584 at this stage.

After maxpooling layer again and the resulting output will be 37x37x128. The the third and fourth convolutional layers a max pooling layer following these two layers, with stride 2 is applied. In the end, it reaches to the maxpooling layer and we have final feature map. We apply droupot of 20% to avoid overfitting. We tested different numbers for avoiding overffiting but 20% was optimal.

The output of the last Pooling Layer is used as input to the Fully Connected Layer at the conclusion of a convolutional neural network. Our dense layer output 4 classes. The activation function is softmax. The outputs are normalized using the softmax function, which converts them from weighted sum values to probabilities that equal to one. Each number in the softmax function's output is interpreted as the likelihood of belonging to each class.

The used optimizer in VGG16 model is Adam optimizer. It is an optimization algorithm that can be used to update network weights iteratively based on training data instead of the traditional stochastic gradient descent procedure. The used loss function is cross entropy. The difference between two probability distributions for a given random variable is measured by cross-entropy, which can be utilized as a loss function while optimizing classification models.

The training strategy consisted of several steps: the first stop is to load the pre-trained model and freeze it, after we made all the model untrainable, we added a droupout and trained only on dense layer. The training was made only for 20 epochs.

After that, unfreeze the model with training rate of 0,0001 and 50 epochs. During training we applied early stopping. Early stopping allows us to stop the model's training early if the parameter I've specified to moni-

tor in early stopping does not increase. In case there is no changes in validation loss, the early stopping will be applied after 5 epochs. It helps to prevent overfitting as well.

The training of the proposed approach is done in two phases. The first one is training on LLOCT dataset and the second one is to test on MSHC dataset with feature maps of the first dataset. In the first training phase, classification model is trained to recognize different multi-class abnormalities such as age related macular degeneration etc. from the first dataset. In the second training phase, we are trying to classify only two different conditions which is multiple sclerosis and healthy controls.

The other five used model architectures presented in the next following tables so the architecture updates during transfer learning can be displayed.

Layer type	Output Shape	Number of parameter
VGG19	(None, 512)	20,024,384
Dense	(None, 512)	2,359,808
DropOut	(None, 512)	0
Dense	(None, 4)	32,772

Table 5: VGG19 Architecture Update

Layer type	Output Shape	Number of parameter
ResNet50	(None, 512)	23,587,712
Dense	(None, 1024)	2,359,808
DropOut	(None, 1024)	0
Dense	(None, 4)	4100

Table 6: ResNet50 Architecture Update

During network training, a callback function was added. The Model checkpoint callback was used to save the network's weights after each epoch and only if the loss function decreased.
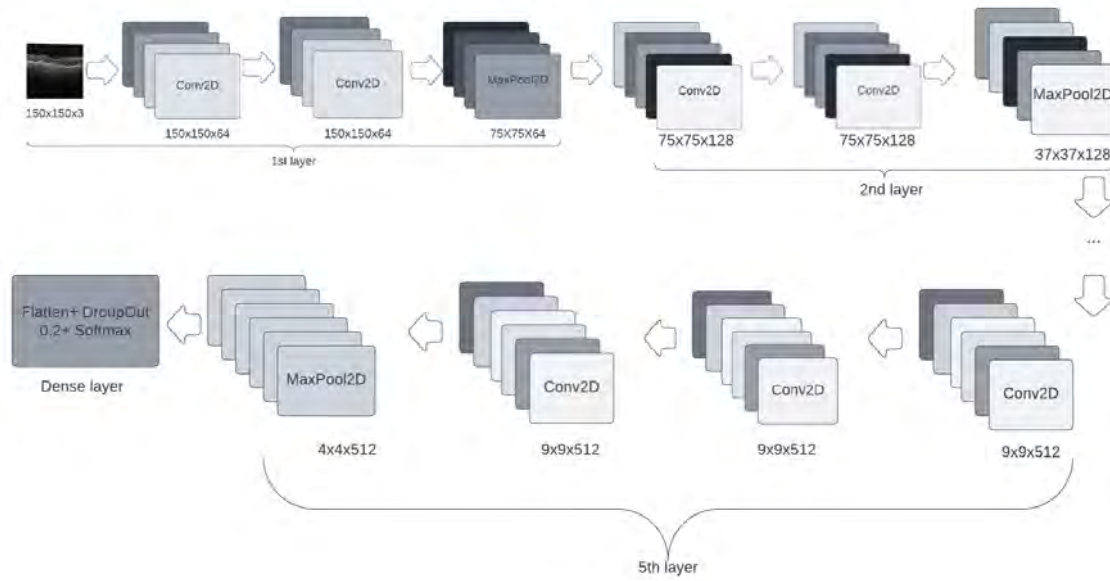
Figure 8: VGG16 Proposed CNN architecture

Layer type	Output Shape	Number of parameter
MobileNet	(None, 1280)	2,257,984
Dense	(None, 1024)	1,311,744
DropOut	(None, 1024)	0
Dense	(None, 4)	4100

Table 7: MobileNet Architecture Update

Layer type	Output Shape	Number of parameter
InceptionV3	(None, 2048)	21,802,784
Dense	(None, 1024)	2,098,176
DropOut	(None, 1024)	0
Dense	(None, 4)	4100

Table 8: InceptionV3 Architecture Update

Layer type	Output Shape	Number of parameter
Xception	(None, 2048)	20,861,480
Dense	(None, 1024)	2,098,176
DropOut	(None, 1024)	0
Dense	(None, 4)	4100

Table 9: Xception Architecture Update

4.5. Evaluation metrics

The classification performance of the models on the test dataset is estimated using four measures in this study. The accuracy of a classifier is the number of times it predicts accurately. The number of correct predictions divided by the total number of forecasts is

Model	Total number of parameters	Trainable Number of parameter
VGG16	14,731,074	16,386
VGG19	20,057,156	32,772
ResNet50	25,689,988	2,102,276
MobileNet	3,573,828	1,315,844
InceptionV3	23,905,060	2,102,276
Xception	22,963,756	2,102,276

Table 10: Parameters

known as accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

The Confusion Matrix is a performance metric featuring a mix of expected and actual results. It is useful for measuring the Recall, Precision, Accuracy, and AUC-ROC curves.

Precision indicates how many of the cases that were correctly predicted turned out to be positive in the end. Precision comes in handy when false positives. The number of true positives divided by the number of anticipated positives is the precision of a label.

$$Precision = \frac{TP}{TP + FP}$$

Recall describes how many of the actual positive cases model properly predicted.

$$Recall = \frac{TP}{TP + FN}$$

It includes a summary of Precision and Recall metrics. When Precision equals Recall, it reaches its peak.

$$F1 = 2x \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.6. Hyperparameters

Deep learning models use a large number of hyperparameters. Tuning the hyperparameters of deep learning models is critical to achieve good predictive performance. The better these hyperparameters are initialized, the faster the model reaches the global minimum. The deep neural networks' hyperparameters, such as batch size, learning rate, and epoch, were fine-tuned to achieve the best results.

The learning rate is a hyperparameter that determines how the network's weights are modified in relation to the loss gradient. Slow convergence is caused by a low learning rate, whereas a high learning rate limits convergence and causes the loss function to oscillate about the minimum. The learning rate for the retinal classification was set on the level of 0.001 for LLOCT scans classification because of the large training set. For the Multiple Sclerosis and Healthy Controls the learning rate was set as The learning rate for 0.00001 as the dataset is much smaller.

Epoch is the number of times the learning algorithm passes over the training set.For all training, the number of epoch is set to 50.

Batch size refers to the number of samples that must run through the model before it is adjusted. All models have a batch size of 10.

Optimizer is a parameter that is used to adjust the parameters for a model. Optimizer used for changing weights and learning rate to reduce the losses. For all the training purpose using Adam optimizer which is a first order gradient based stochastic optimization process. The Adam optimizer is given by following equations:

$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ (1)
$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$ (2)

Where m and v are moving averages, g is the gradient on the current mini-batch, and betas are the algorithm's newly introduced hyper-parameters.

Loss function is used to compute the distance between the true and the model's predicted labels.

4.7. Implementation

This work was implemented using Python programming language. For the second dataset, image preparation was performed in Matlab. The models were implemented in Keras, using a GPU NVIDIA Tesla P100. The CNN architectures employed in this study were VGG-16, VGG-19, ResNet50, MobileNet, Inception V3 and Xception, which were all found in Keras Applications. When a model was created, ImageNet weights were downloaded automatically during model installation.

5. Results

In this section, the experimental results are presented for both LLOCT and MSHC datasets. The results will be presented in two parts, the first one is the results on LLOCT dataset, the second one is on MSHC dataset. The suggested strategy performance was assessed using standard classification measures. F1-score was employed to obtain unbiased findings in the imbalanced situation (particularly with the LLOCT dataset).

5.1. Results with the Large Labeled Optical Coherence Tomography (OCT) Images dataset

The confusion matrix, which contains the correct and incorrect classification results for each model, is used to calculate the performance. From Figure 9 to Figure 14 the confusion matrices are presented for the first dataset.
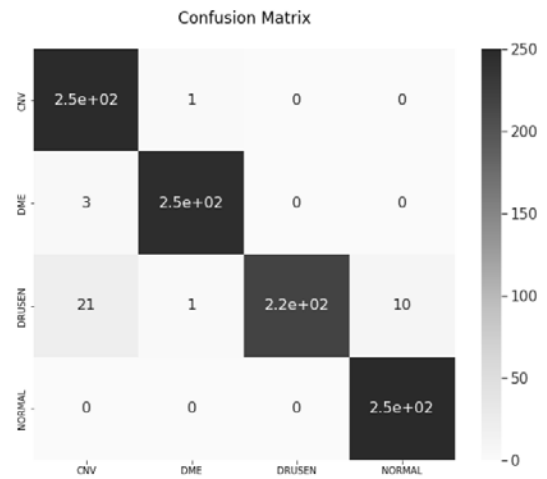
Figure 9: Confusion Matrix VGG16
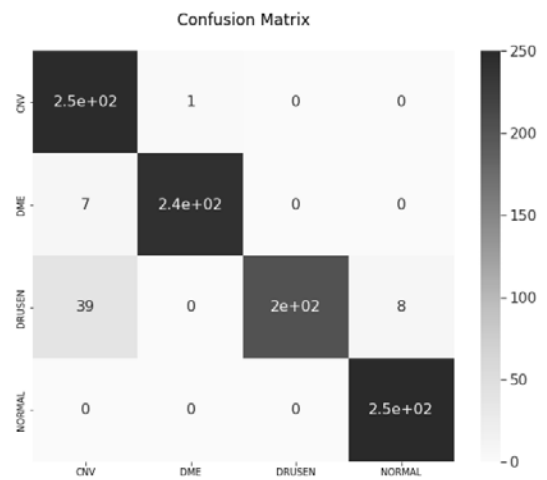
Figure 10: Confusion Matrix VGG19

The performance of models VGG16, VGG19, ResNet50, MobileNet, InceptionV3 and Xception on
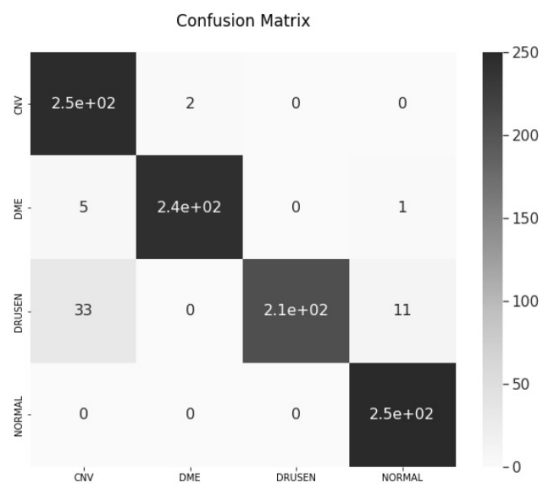
Figure 11: Confusion Matrix ResNet50

Figure 12: Confusion Matrix MobileNet
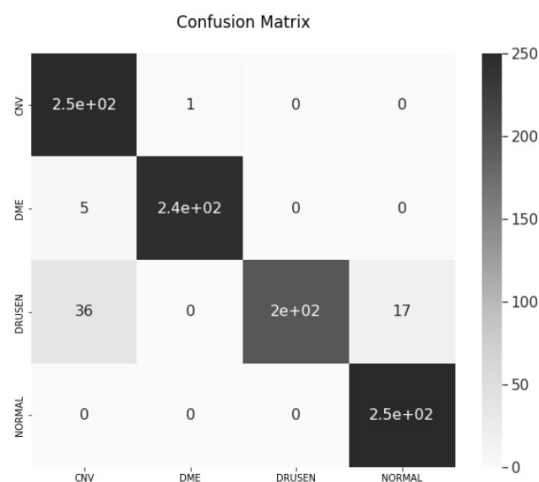
Figure 13: Confusion Matrix InceptionV3

Figure 14: Confusion Matrix Xception

the Large Labeled Optical Coherence Tomography (OCT) Images dataset was evaluated and compared according to accuracy criteria. Figure 15 demonstrates a comparison of achieved accuracy with LLOCT dataset. The best performance was achieved by VGG16 with accuracy of 0.96 and the worst result by MobileNet with accuracy of 0.85. Furthermore, because VGG16 was the best-performing model in the group, fine-tuning comparisons were performed with it.

The performance on the LLOCT dataset is shown in Table 11.

5.2. Results on The Multiple Sclerosis and Healthy Controls dataset

Table 12 displays the obtained results on the second dataset. VGG16 achieved the best model performance so for the classification prediction on the second dataset was do by using the weight of the best model. The result of achieved accuracy on MSHC dataset is 55%.

The suggested method of transfer learning was tested using 2 publicly available datasets in two different types of retinal diseases. While VGG16 performed best on the first dataset, the proposed strategy was also tested with VGG-19, MobileNet, ResNet50, InceptionV3, and Xception. The best model was used for multiple sclerosis cases.

6. Results and Discussion

The aim of this project was the investigation of retinal diseases by analyzing the OCT scans and to realize a transfer learning algorithm for processing retinal images so that we can diagnose various important pathologies in each case accurately. The goal of these types of methodologies is to develop a tool that can assist in multi-classification and may also be useful to ophthalmologists.
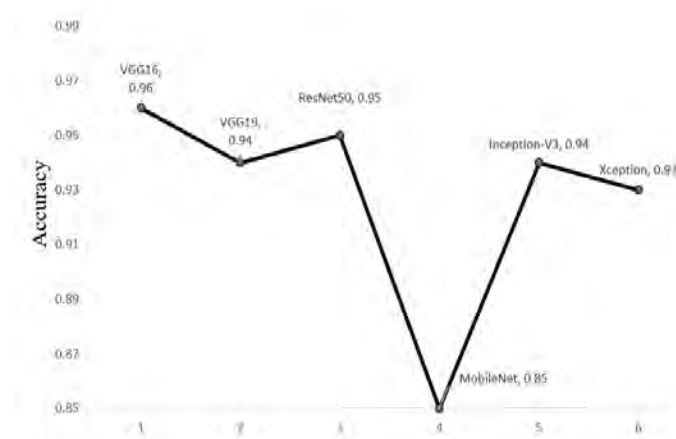
Figure 15: Comparison of achieved accuracy on the the Large Labeled Optical Coherence Tomography (OCT) Images dataset

Class	Precision	Recall	F-1 Score
VGG16			
CNV	0.91	1.00	0.95
DME	0.99	0.99	0.99
DRUSEN	1.00	0.87	0.93
NORMAL	0.96	1.00	0.98
VGG19			
CNV	0.84	1.00	0.91
DME	1.00	0.97	0.98
DRUSEN	1.00	0.81	0.90
NORMAL	0.97	1.00	0.98
ResNet50			
CNV	0.85	1.00	0.92
DME	0.99	0.98	0.99
DRUSEN	1.00	0.80	0.95
NORMAL	0.96	1.00	0.98
MobileNet			
CNV	0.74	0.98	0.84
DME	0.99	0.92	0.96
DRUSEN	0.99	0.51	0.67
NORMAL	0.82	1.00	0.90
InceptionV3			
CNV	0.86	1.00	0.92
DME	1.00	0.98	0.99
DRUSEN	1.00	0.79	0.88
NORMAL	0.94	1.00	0.97
Xception			
CNV	0.82	1.00	0.90
DME	1.00	0.98	0.99
DRUSEN	0.99	0.76	0.86
NORMAL	0.96	1.00	0.98

Table 11: Classification Results on the the Large Labeled Optical Coherence Tomography (OCT) Images dataset

Convolutional neural networks were tested in this study for predicting retinal abnormalities using OCT retinal scans. The performance of VGG16, VGG19, ResNet50, MobileNet, InceptionV3 and Xception con-
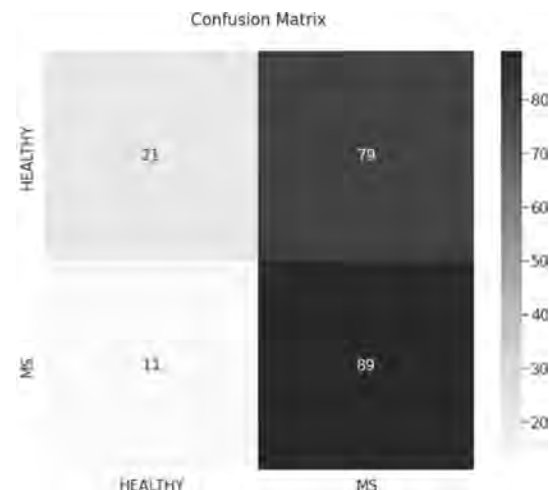
Figure 16: Confusion Matrix VGG16 on The Multiple Sclerosis and Healthy Controls dataset

Class	Precision	Recall	F-1 Score
Multiple Sclerosis	0.53	0.89	0.66
Healthy Controls	0.66	0.21	0.32

Table 12: Classification Results of VGG16 on The Multiple Sclerosis and Healthy Controls dataset

volutional neural networks in publicly available OCT dataset with different pathologies is evaluated and discussed based on our methodology.

First, the intended goal was to do testing across four different retinal conditions on the first dataset using six different pre-trained models and after to apply the best model on the second dataset for classification of another eye condition . With a 96 % accuracy, the suggested framework was able to achieve best performance in transfer learning on LLOCT dataset. In the work, we dealt with an issue of decreased performance on the second dataset. Table 12 demonstrates that obtained results are significantly decreased. The VGG16 on MSHC

dataset achieved a 55% classification accuracy. The considerable difference in these scores is partly due to the fact that the difference between the two datasets is probably more than expected.

The important component of this work is the confirmation that transfer learning may be used to classify OCT images of choroidal neovascularization (CNV); diabetic macular edema (DME); drusen; normal classes using suitable CNN-based models with associated algorithm hyperparameters. According to the findings of this study, the training set in retinal classification with the first dataset performed better, however the training in the second dataset classification with classes associated with multiple sclerosis performed worse. Because classification outcomes are strongly dependent on the used data.

The possible causes of that low performance can be difference in the images between two datasets, a few MS cases since the second dataset is much smaller, bigger differences in the pathology than expected previously. It's possible that the problem with performance on the second dataset is due to a lack of samples. In future development, the performance could be enhanced with a larger training dataset is expected to improve the model's accuracy.

7. Future work

As a further step for future work, we expect to receive additional data from the University of Padova (Italy). The laboratory is focused on studying degenerative neurological diseases such as multiple sclerosis and dementia.

The future work will be focused on detecting illness such as Multiple Sclerosis and the classification of biomarkers. The continual learning aim combines and displays complex representational connections between prior and new knowledge. Continual learning is used for networks that can continuously acquire knowledge across multiple classes without having to retrain. The future work will include further experiments with neurological diseases and its biomarkers in eyes.

Acknowledgments

References

Arabi, P.M., Krishna, N., Deepa, N.V., Ashwini, V., Prathibha, H.M., 2017. A comparision of oct and retinal fundus images for age-

related macular degeneration, in: 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1–5. doi:10.1109/ICCCNT.2017.8204107.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258.

Eladawi, N., Elmogy, M., Ghazal, M., Helmy, O., Aboelfetouh, A., Riad, A., Schaal, S., El-Baz, A., 2018. Classification of retinal diseases based on oct images. Frontiers in Bioscience-Landmark 23, 247–264.

Fang, L., Wang, C., Li, S., Rabbani, H., Chen, X., Liu, Z., 2019. Attention to lesion: Lesion-aware convolutional neural network for retinal optical coherence tomography image classification. IEEE Transactions on Medical Imaging 38, 1959–1970. doi:10.1109/TMI.2019.2898414.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

He, Y., Carass, A., Liu, Y., Jedynak, B.M., Solomon, S.D., Saidha, S., Calabresi, P.A., Prince, J.L., 2019a. Deep learning based topology guaranteed surface and mme segmentation of multiple sclerosis subjects from retinal oct. Biomedical optics express 10, 5042–5058.

He, Y., Carass, A., Liu, Y., Jedynak, B.M., Solomon, S.D., Saidha, S., Calabresi, P.A., Prince, J.L., 2019b. Fully convolutional boundary regression for retina oct segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 120–128.

He, Y., Carass, A., Liu, Y., Jedynak, B.M., Solomon, S.D., Saidha, S., Calabresi, P.A., Prince, J.L., 2020. Structured layer surface segmentation for retina oct using fully convolutional regression networks. Medical Image Analysis , 101856.

He, Y., Carass, A., Solomon, S., Saidha, S., Calabresi, P., Prince, J., 2019c. Retinal layer parcellation of optical coherence tomography images: Data resource for multiple sclerosis and healthy controls. Data in Brief 22, 601–604. doi:10.1016/j.dib.2018.12.073. funding Information: This work was supported by the NIH under NEI grant R01-EY024655 (PI: J.L. Prince) and NINDS grant R01-NS082347 (PI: P.A. Calabresi). Publisher Copyright: © 2019.

He, Y., Carass, A., Solomon, S.D., Saidha, S., Calabresi, P.A., Prince, J.L., 2019d. Retinal layer parcellation of optical coherence tomography images: Data resource for multiple sclerosis and healthy controls. Data in brief 22, 601–604.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 .

Karri, S.P.K., Chakraborty, D., Chatterjee, J., 2017. Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration. Biomed. Opt. Express 8, 579–592. doi:10.1364/BOE.8.000579.

Kermany, D.S., Goldbaum, M., Cai, W., Valentim, C.C., Liang, H., Baxter, S.L., McKeown, A., Yang, G., Wu, X., Yan, F., Dong, J., Prasadha, M.K., Pei, J., Ting, M.Y., Zhu, J., Li, C., Hewett, S., Dong, J., Ziyar, I., Shi, A., Zhang, R., Zheng, L., Hou, R., Shi, W., Fu, X., Duan, Y., Huu, V.A., Wen, C., Zhang, E.D., Zhang, C.L., Li, O., Wang, X., Singer, M.A., Sun, X., Xu, J., Tafreshi, A., Lewis, M.A., Xia, H., Zhang, K., 2018a. Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 172, 1122–1131.e9.

Kermany, D.S., Goldbaum, M., Cai, W., Valentim, C.C., Liang, H., Baxter, S.L., McKeown, A., Yang, G., Wu, X., Yan, F., et al., 2018b. Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 172, 1122–1131.

Khan, A.M., Hassan, T., Akram, M.U., Alghamdi, N.S., Werghi, N., 2022. Continual learning objective for analyzing complex knowledge representations. Sensors 22. doi:10.3390/s22041667.

Khan, S.M., Liu, X., Nath, S., Korot, E., Faes, L., Wagner, S.K., Keane, P.A., Sebire, N.J., Burton, M.J., Denniston, A.K.O., 2021. A global review of publicly available datasets for ophthalmological imaging: barriers to access, usability, and generalisability. The

Lancet. Digital health 3 1, e51–e66.

Landau, K., Kurz-Levin, M., 2011. Retinal disorders. Handbook of clinical neurology 102, 97–116.

Li, F., Chen, H., Liu, Z., Zhang, X.d., Jiang, M.s., Wu, Z.z., Zhou, K.q., 2019. Deep learning-based automated detection of retinal diseases using optical coherence tomography images. Biomedical Optics Express 10, 6204. doi:10.1364/BOE.10.006204.

Litjens, G.J.S., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J., van Ginneken, B., Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Medical image analysis 42, 60–88.

Luo, Y., Xu, Q., Jin, R., Wu, M., Liu, L., 2021. Automatic detection of retinopathy with optical coherence tomography images via a semi-supervised deep learning method. Biomed. Opt. Express 12, 2684–2702. doi:10.1364/BOE.418364.

Naz, S., Hassan, T., Akram, M.U., Khan, S.A., 2017. A practical approach to oct based classification of diabetic macular edema, in: 2017 international conference on signals and systems (ICSigSys), IEEE. pp. 217–220.

Organization, W.H., 2019. World report on vision. World Health Organization.

Petzold, A., de Boer, J.F., Schippling, S., Vermersch, P., Kardon, R., Green, A., Calabresi, P.A., Polman, C., 2010. Optical coherence tomography in multiple sclerosis: a systematic review and meta-analysis. The Lancet Neurology 9, 921–932.

Rong, Y., Xiang, D., Zhu, W., Yu, K., Shi, F., Fan, Z., Chen, X., 2019. Surrogate-assisted retinal oct image classification based on convolutional neural networks. IEEE Journal of Biomedical and Health Informatics 23, 253–263. doi:10.1109/JBHI.2018.2795545.

Sezgin, M., Sankur, B., 2004. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic imaging 13, 146–165.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. Journal of big data 6, 1–48.

Simonyan, K., Zisserman, A., 2014a. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

Simonyan, K., Zisserman, A., 2014b. Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 .

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.

Triwijoyo, B.K., Budiharto, W., Abdurachman, E., 2017. The classification of hypertensive retinopathy using convolutional neural network. Procedia Computer Science 116, 166–173.

Wang, D., Wang, L., 2019. On oct image classification via deep learning. IEEE Photonics Journal 11, 1–14. doi:10.1109/JPHOT.2019.2934484.

Yang, S., Xiao, W., Zhang, M., Guo, S., Zhao, J., Shen, F., 2022. Image data augmentation for deep learning: A survey. arXiv preprint arXiv:2204.08610 .

Zhang, M., Wang, J.Y., Zhang, L., Feng, J., Lv, Y., 2019. Deep residual-network-based quality assessment for sd-oct retinal images: preliminary study, in: Medical Imaging 2019: Image Perception, Observer Performance, and Technology Assessment, SPIE. pp. 269–274.
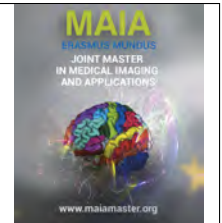
Breast Mass Detection and Classification Using Transfer Learning

Marya Ryspayeva, Mario Molinara

Department of Electrical and Information Engineering, University of Cassino and Southern Lazio, Cassino, Italy

Abstract

X-ray mammography is the gold standard for diagnosing early signs of breast cancer, while Artificial Intelligence is a modern method that enables detecting suspicious lesions and classifying them in terms of malignancy. This thesis aimed to investigate mass detection with three Transfer Learning settings in the early screening and mass classification in a large-scale OPTIMAM (OMI-DB) dataset with 6000 cases and extracted more than three thousand images with masses in the mammograms of Hologic manufacturer. The methodology of the detection step is to train the RetinaNet architecture of three ResNet50, ResNet101, and ResNet152 backbones with three types of the initialization by ImageNet, COCO weights and from scratch. The dataset was pre-processed to generate two types of input with entire mammograms and patches, which are stated as the first and the second approaches. Received results show that in the first approach, RetinaNet of ResNet50 backbone with ImageNet weights and ResNet152 with the same weights performed 0.944 and 0.959 True Positive Rate (TPR) at 0.84 False Positive Per Image (FPPI), respectively, RetinaNet with ResNet50 and COCO weights reached 0.938 TPR at 0.84 FPPI. We showed that RetinaNet with ResNet152 initialized ImageNet weights achieved state-of-the-art results in the first approach with entire mammograms. In the classification step, we applied the Transfer Learning approach with fine-tuning by adding L2-regularization and class weights to balance class distribution in the datasets. The classification step demonstrated high precision, recall, F1-score and accuracy.

Keywords: Breast cancer, Mass detection, Mass classification, Transfer learning, OPTIMAM, OMI-DB, Artificial Intelligence

1. Introduction

Radiologists accept that X-ray mammography has become the gold standard of early breast cancer diagnosis for women. Usually, early screening is taken from 40 to 70 age every two years (Sechopoulos et al., 2021). However, last two years, several countries reported decreasing screening rates due to the Coronavirus restrictions. The pandemic has affected all areas of human life, especially the medical field, where the patient was forced to choose between not getting infected with the coronavirus and timely treatment of the disease. It also affected breast cancer due to quarantine, increased patient service time, and routine procedures. Additional measures such as disinfection after each patient, restriction of people and staff in waiting and medical rooms, and redeployment of medical resources have also caused a drop in the number of screenings for early detection of breast cancer (Monticciolo et al., 2021).

Many countries claimed a significant drop in the screening test rate during the pandemic. For instance, in Italy, Battisti et al. (2022) evaluated it at around 40%, French radiologists determined it by 10% (Le Bihan Benjamin et al., 2022), Catalonia region in Spain reported a decrease from 21% to 37%, and almost 20% in the first year of the pandemic (Ribes et al., 2022). Moreover, Italian researchers observed a decline of up to 15% of the patients participating in breast cancer screening (Battisti et al., 2022). The number of routine procedures in the United Kingdom (UK) also declined by 40% in 2020 compared to the previous year (Gathani et al., 2022). 26% of Spanish patients with the already diagnosed disease are exposed to high lethal risk and malignancy in the next few years (Ruiz-Medina et al.,

2021). Scientists specify that after two years of the pandemic, there is a gap in breast cancer detection despite the return of regular screenings and the growth of undetected cancer (Le Bihan Benjamin et al., 2022), (Ribes et al., 2022).

Researching breast cancer preserves its importance, especially during the pandemic and other times. Successful investigations can decrease mortality and disease severity through early diagnosis and efficient treatment. Modern methods can detect and diagnose tumors by Artificial Intelligence (AI). AI can help speed up the detection and diagnosis of disease to overcome such difficulties as physician fatigue, diagnostic errors, and time-consuming annotation, which aggravated due to COVID in recent years and catch up on the delay due to the pandemic. Physicians worldwide collaborate with researchers to develop automatic systems for abnormalities detection, classification, segmentation, and other tasks in breast cancer diagnosis. Such systems could help avoid human and diagnostic errors, manual reading, and overlooked lesions due to radiologists' fatigue. However, there are challenges in preparing data for the AI methods, such as artifacts, noise, and manual labeling by experts requesting medical knowledge and time.

Many algorithms in the medical domain were developed using Transfer Learning as one of the AI methods and reached state-of-the-art results. Breast cancer disease has been investigated by many researchers who applied different methods, for instance, Convolutional Neural Networks (CNN) to detect abnormalities in mammograms (Elia et al., 2008), (Bria et al., 2016), (Marrocco et al., 2005), (Savelli et al., 2020). This thesis proposes a computer-aided diagnosis system (CAD) to detect and classify masses as benign and malignant on the full-field digital mammograms (FFDM) of the OPTIMAM (OMI-DB) dataset using Transfer Learning in order to speed up the screening process.

This study aimed to investigate how the Transfer Learning approach affects the performance of a system for breast cancer detection and classification. In particular, it was considered a detector based on RetinaNet with two different Transfer Learning settings (ImageNet and COCO) and trained it from scratch. The classification is based on fine-tuning Transfer Learning by adding L2-regularization.

Moreover, two approaches to the input to the RetinaNet model were compared. The first approach was based on using the pre-processed image by downsampling and cropping the entire FFDM as the input. In the second approach, we generated patches of 500 × 500 pixels from the entire FFDM due to the large size of the original mammograms. It has been proved

that pre-processing step can improve detection results; however, it was also investigated that CNN architectures detect abnormalities efficiently without pre-processing step (Bria et al., 2018), (Marchesi et al., 2017). Both approaches can be accepted, and the pre-processing step was applied to Transfer Learning with pre-trained models in detection and classification tasks in this research. Masses were predicted using the RetinaNet model with different backbones trained with weights. Also, we compared the results with fitting the model from scratch. Mean Average Precision (mAP) was chosen as the primary metric to evaluate models during the training process on the validation subset. Test subsets were evaluated with a True Positive Rate (TPR) at False Positive Per Image (FPPI), which helped us compare received results with a state-of-the-art. The classification step used an approach of Transfer Learning, unfreezing all layers, fine-tuning by adding L2-regularization layers. The contributions of this work are as follows:

1. We detected masses using the RetinaNet model with three types of the backbones ResNet50, ResNet101, and ResNet152 with two approaches of entire mammograms and patches. The number of the models is 18 in two approaches together.

2. To the best of our knowledge, this thesis is the first where three backbones were trained with two types of weights (ImageNet and COCO) and from scratch.

3. The model RetinaNet was applied to OMI-DB dataset with two approaches of input: entire mammograms and patches while other authors used only one of the methods in their research.

4. We implemented two tasks of the detection and classification in this research. Predicted bounding boxes were classified into benign and malignant tumors using the Transfer Learning method.

5. We classified highly-imbalanced datasets with the distribution of the minority class of 10% versus 90% of the majority class.

The thesis is organized as follows: "State-of-the-art" describes the current mass detection and classification situation, considers methods, datasets, and achieved results by other researchers. The following section, "Materials", includes information about the employed OMI-DB dataset, image distribution, and parameters, pre-processing for detection and classification steps. "Methods" contains two approaches of the detection methods with their details and classification steps. "Results" presents the achieved results in terms of the metric and comparison with the state-of-the-art. Finally, we discuss the research value, its usefulness, and obtained results.

2. State of the art

Majority of papers directed at breast cancer highlight only detection or classification steps simultaneously. A rare paper can be found which implemented two tasks together and predicted abnormalities transferred to the classification model. This section presents two subsections of state-of-the-art mass detection and classification using Transfer Learning over the past few years.

2.1. Breast mass detection

Agarwal et al. (2020) characterize their paper as the first paper, in which the AI method Deep Learning was applied to the OMI-DB dataset (Halling-Brown et al., 2021). The authors received a subset of the large-scale OMI-DB dataset with 4750 cases, where 2145 cases were with cancer from two manufacturers of Hologic Inc. (OMI-H) and General Electric (OMI-G). From these cases, they extracted images with masses, and the total numbers in two selected subsets were 2042 positive with 842 normal cases of OMI-H and 103 with 104 cases of OMI-G. Moreover, an additional dataset INbreast was utilized with 50 positive and 65 normal cases (Moreira et al., 2012). However, the amount of benign and malignant images caused an imbalance distribution of 485 versus 3048, respectively. As the pre-processing step in the OMI-H and OMI-G subsets, original mammograms were down-sampled due to the large size of the original FFDMs, normalized, and rescaled to 8 bits. Agarwal et al. (2020) marked that they used the entire mammogram as an input to the algorithm. As a primary model, the authors applied Faster R-CNN, a two-step model with regression and classification networks (Girshick, 2015). InceptionV2 backbone pre-trained on the COCO dataset was chosen as a feature extractor. Region Proposal Network (RPN) predicted bounding boxes of all possible objects, evaluated with a confidence score of an overlapping predicted object with groundtruths. The RPN consists of two classification and regression networks, where a predicted bounding box is classified as mass or non-mass, in other words, foreground or background. In terms of the classification results of mass, the coordinates are redefined in the regression network with the probability of closeness to the groundtruth. OMI-H dataset was trained on the pre-trained COCO weights by Faster R-CNN with 0.87 TPR at 0.84 FPPI on the testing subset. Furthermore, to predict masses in OMI-G and INbreast datasets, the trained model on OMI-H, was fine-tuned and demonstrated 0.91 TPR at 1.70 FPPI in OMI-G and 0.99 TPR at 1.17 FPPI in INbreast datasets.

Sulaiman et al. (2021) as well as Agarwal et al. (2020) investigated breast cancer with two mass detection and classification tasks by applying Faster R-CNN. However, as materials, the authors chose two datasets of MIAS and the Curated Breast Imaging Subset of Digital Database for Screening Mammography (CBIS-DDSM) (Sawyer-Lee et al., 2016). According to Agarwal et al. (2020), the whole FFDM was used as an input to the detection step. Both datasets were augmented and pre-processed to eliminate artifacts on the images by multi-threshold peripheral equalization and resized to 256×256 pixels. MatConvNet as feature extractor showed the top 97.04% accuracy.

Lotter et al. (2021) developed the classification and detection CAD system, which works with 2D and 3D mammograms. As datasets, DDSM (Mammography et al., 2001) and OMI-DB were used as 2D data of FFDMs and characterized as strongly-annotated by experts, unlike 3D images of weakly-labeled Digital Breast Tomosynthesis due to a high number of the slices. Additionally, three private datasets were used for training. In the OMI-DB datasets, images of two manufacturers, General Electric and Hologic Inc., were extracted. The OMI-DB dataset consists of 5233 positive studies and 16887 negative studies. The algorithms started from the first step of the patches classification of 275×275 pixels from two datasets. As a result, it generated two million patches of the balanced distribution of positive and negative patches. These patches were classified with a pre-trained ResNet50 model with ImageNet weights to four abnormalities and no lesion, which were input to the RetinaNet detection model. The results of the paper present 0.963 AUC in the OMI-H dataset.

2.2. Breast mass classification

The classification task is investigated more frequently in breast cancer research than the detection task. It is connected with the complexity of the architectures, training process, and the need for computing power to predict bounding boxes as a regression problem than to perform binary or multiclass classification. The majority of papers limelight classification reaching standalone results with different models and approaches of Transfer Learning.

Valerio et al. (2019) aggregated two datasets DDSM and the VIENNA dataset, to MAMMOSET, with 3339 total mammograms. Unlike other papers, the authors solved an 11-class classification task according to the Breast Imaging Reporting and Data System (BI-RADS). Before data augmentation, the distribution of the classes in the combined MAMMOSET was highly-imbalanced. The total number of the images summed up more than 10 thousand, which were balanced by data augmentation. The paper compared handcrafted features with traditional classifiers of Machine Learning

and in-depth features of Transfer Learning. Researchers concluded that the best performance was demonstrated in-depth features with the CNN model of the augmented dataset, and the NASNet-Large model demonstrated a maximum of 93.40% accuracy.

In contradistinction to Valerio et al. (2019), Falconi et al. (2020) predicted 6-class classification according to the BI-RADS and generated patches of 399 x 399 pixels of the INbrest dataset's FFDMs. The authors compared three pre-trained models, where NASNet Mobile used the Transfer Learning approach. At the same time, two VGG16 and VGG19 were fine-tuned by adding Global average pooling directly before the dropout layer. In contrast, in Transfer Learning, Global average pooling came before the last fully-connected layer with the following dropout layer. Fine-tuned models showed 0.885 and 0.909 accuracies, respectively.

Yu and Wang (2019) detected abnormalities on the mammograms with the pre-trained model ResNet18 of Transfer Learning. Material for the algorithms was a public mini Mammographic Image Analysis Society (MIAS) dataset (Suckling et al., 2015) with 322 images from 161 patients with breast images where physicians labeled bounding boxes for the abnormalities. The authors used a patch-based approach to generating regions of interest. Images without abnormalities underwent pre-processing with morphological operations and binarization with a selection of the largest area, and positive images were cropped out according to the labeled ground truths. They compared three ResNet models such as ResNet18, ResNet50, and ResNet101, as feature extractors and froze layers before the last fully-connected (FC) layer, and retrained three layers of FC, softmax, and classification layer. Achieved results showed that ResNet18 classified mammograms with 95.91% of mean accuracy.

Alruwaili and Gouda (2022) selected the MIAS dataset as materials to classify mammograms as benign and malignant. As standard pre-processing steps of data augmentation, image enhancement, rescaling, and normalization were applied to exclude overfitting and increase performance. NASNet-Mobile and ResNet50 were chosen as pre-trained models in the Transfer Learning approach with 89.5% accuracy of the residual network. Khamparia et al. (2021) also proved that pre-trained models, for instance, VGG16, by fine-tuning and adding data augmentation, could help reach 86.9% accuracy in the classification of pathological and non-pathological samples.

Several papers developed hybrid models with classifiers of Machine Learning (Mahmood et al., 2021) or other Deep Learning Networks (Altaf, 2021).

Mahmood et al. (2021) extracted features using six pre-trained models VGG19, VGG19, GoogLeNet, MobileNetV2, ResNet50 and DenseNet121. Extracted features of each pre-trained model were fitted with the Support Vector Machine (SVM) classifier of Machine Learning. As a result, VGG19+SVM showed 93.5% accuracy. Altaf (2021) deployed a hybrid model based on Pulse-Coupled Neural Networks (PCNN) and Transfer Learning pre-trained models. The authors stated that researchers could train entire images without the segmentation and pre-processing to generate image signatures and obtain high results. About data, Mahmood et al. (2021) merged three datasets, trained, and showed an accuracy of 98.9% with the GoogLeNet model. In contrast, Altaf (2021) with PCNN, which generated image signatures invariant to acquisition quality and transformations, showed 98.72% of accuracy.

3. Materials

The challenge in the medical domain is the availability of large datasets to train Deep Learning models effectively. Many researchers combine several public datasets to operate with more data and obtain robust algorithms based on state-of-the-art analysis. In this thesis, we use a large-scale OMI-DB dataset as Agarwal et al. (2020) and Lotter et al. (2021). Additionally, these authors tested the algorithms on public datasets and subsets of different manufacturers. Unlike them, we analyzed only the OMI-H subset and tested our algorithms on it.

The OMI-DB dataset collected over 2.5 million FFDMs during the UK's National Health Service Breast Screening Program from more than 170 thousand women (Halling-Brown et al., 2021). However, most papers indicated a various number of cases, images distribution, and manufacturers. Such differences suggest that other authors and we were provided with distinct subsets due to the high number of images in the original large-scale mammography dataset. The OPTIMAM steering committee considers requests for data sharing depending on the various research aims. Our dataset includes 6000 cases with 148,461 processed and unprocessed FFDMs. A case is assumed to be a patient examined for some time, and each case contains studies with time intervals where the examination images were kept. The annotation information describes modality, breast view, screening date, abnormalities type, expert's labeling in the presence of abnormalities and malignancy (benign or malignant), and other information regarding the screening, patient, and image parameters. The labeling was provided as bounding box coordinates where one or several lesions were located.
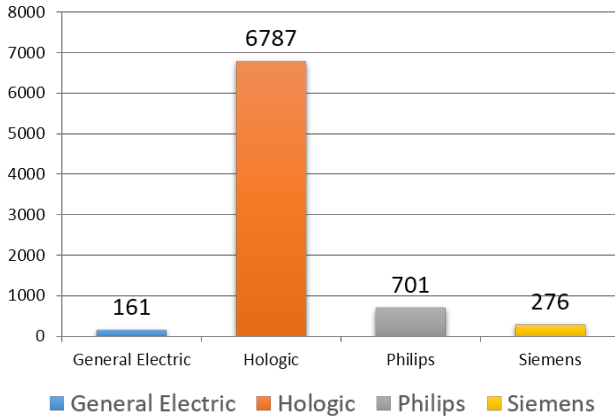
Figure 1: Manufacturers' distribution in the OMI-DB dataset with images number which is marked as abnormal.

Figure 2: Example of mammogram views: a) Right CC b) Left CC c) Right MLO d) Left MLO.

Figure 3: The workflows of the pre-processing steps for the first and the second approaches.

The ratio of the manufacturers in the OMI-DB is shown in Figure 1, demonstrating images with any abnormalities of four manufacturers. As it is seen that the dataset is highly imbalanced in terms of the manufacturers, and General Electric, Philips, and Siemens were presented as a minority, hence we selected only 6787 images of Hologic Inc.

All images in the OMI-H subset are performed in two projections for left and right breasts: the mediolateral oblique (MLO) of 2560×3328 pixels and craniocaudal (CC) of 3328×4048 pixels (Figure 2). The OMI-H contains samples with various abnormalities such as architectural distortion, calcification, focal asymmetry, mass, and others. However, we considered only a mass abnormality and the final number of the positive images with masses made up 3524 images and 4100 negative images without masses. All images were split into training, validation, and test subsets with 70%, 10%, and 20%, respectively. We converted original grayscale DICOM mammograms with a 16-bit range of intensity to png with 8 bits.

As it was analyzed in the literature review, Agarwal et al. (2020) used entire mammograms as inputs to the detection algorithm, while Lotter et al. (2021) developed patch-based architecture with 275 pixels in height and width. In this thesis, we conducted the research of two approaches, where entire mammograms and extracted patches are performed as inputs to the models.

The first approach requires taking all images as they are. Due to the mentioned original size of MLO and CC views, AI models can not process such huge images as it demands enormous computational resources. We followed the same pre-processing steps as Agarwal et al. (2020). However, Agarwal et al. (2020) down-sampled all images to 200 micrometers; we chose the lowest res-

olution and down-sampled them to 70 micrometers to keep the proportions and coordinates of the bounding boxes. Moreover, we cropped out the breast area according to segmented masks obtained from unprocessed mammograms with a high contrast of the breast and air. As a result, the total number of benign and malignant samples was composed of 362 and 3162 images, respectively. The workflows of the pre-processing steps for both approaches are performed in Figure 3.
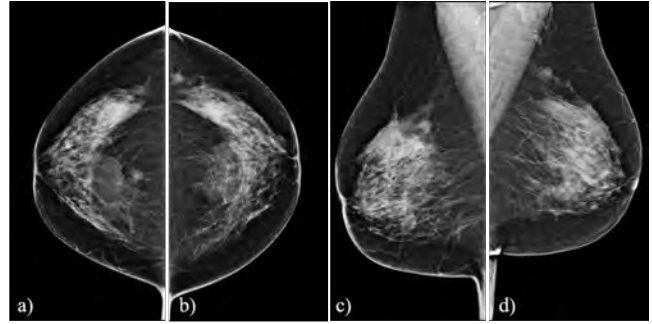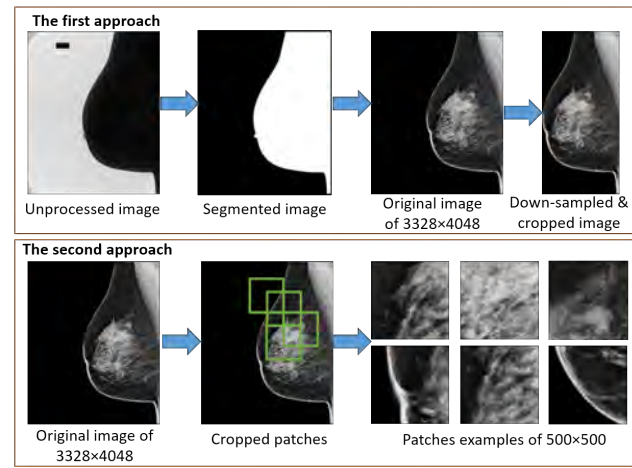
The second approach is based on the extracted patches from the original images. A sliding window creates patches with the horizontal and vertical step of 250 pixels that runs through the entire image to generate patches. Each patch is 500×500 pixels with bounding box coordinates, and the total number of patches is 500616. In this thesis, we consider only positive images. Hence, we selected patches only containing masses coordinates, and the total number of the images made up 2690 benign versus 25396 malignant. The splitting ratio to train, validation and the test subsets was left as in the first approach. The workflow of the pre-processing steps for the second approach is also shown in Figure 3.
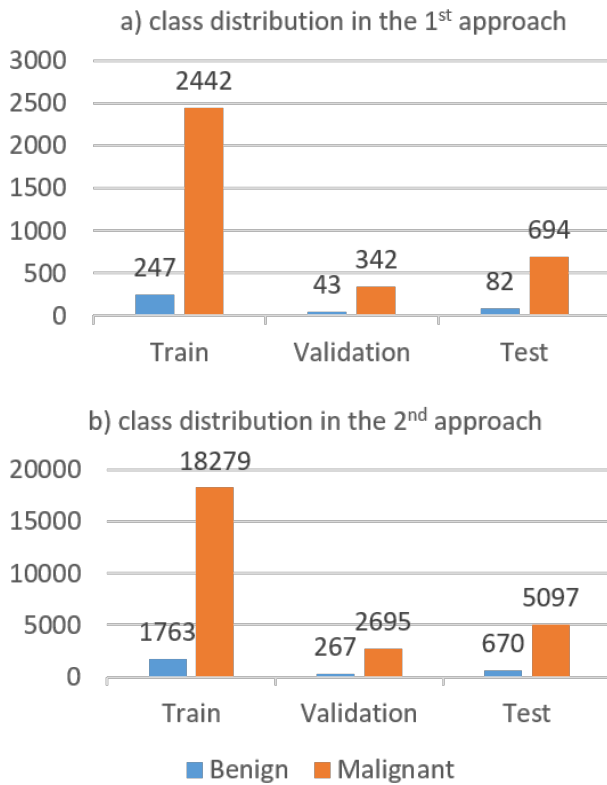
Figure 4: Benign and malignant images are distributed in training, validation, and test subsets of a) the first and b) the second approaches.

After getting the results of the detection model, predicted bounding boxes had to be classified as benign and malignant. We cropped predicted bounding boxes to obtain only masses, however, their size varies significantly. Some images labeled with several masses, which all were cropped out from the images and patches and pre-processed. The pre-processing step was directed to increase the neighborhood pixel space of the objects to 250 pixels in height and width. If the height or width equals to 250 and more, we left them as they are. As a result, the minimal size of the image with masses inside is 250×250 pixels. The training and validation subsets distributions are the same as the detection step and it underwent the same pre-processing step by increasing the size of around the mass to 205 pixels as the predicted masses. Figure 4 demonstrates the distribution of the classes inside two datasets of two approaches in terms of the training, validation, and test for the classification task.

4. Methods

State-of-the-art shows that Transfer Learning as a method of AI can reach high results in the detection

and classification of the medical domain. Especially in the medical domain, that distinguishes significantly from other datasets having specific images and objects. Transfer Learning recommended itself to be one of the leading AI methods in this field, transferring knowledge from large-scale datasets of ImageNet (Deng et al., 2010) and COCO to small and specific data (Lin et al., 2014). Transferring the knowledge from one domain to another saves training time and computational resources and gives good results.

ImageNet is a large-scale database with 14 million images with 1000 categories. During ImageNet Large Scale Visual Recognition Challenge, developers trained models on the ImageNet datasets and shared their models with ImageNet weights (Deng et al., 2010). On the other hand, COCO weights were trained using COCO datasets with 328 thousand images of 80 categories (Lin et al., 2014). To get the weights, ResNet50 was fitted on the COCO dataset. The pre-trained weights help provide a starting point to the own training model and reach convergence in fewer epochs.

4.1. Breast mass detection

As a detection method, Agarwal et al. (2020) and Sulaiman et al. (2021) utilized the Faster R-CNN model with two regression and classification networks with different datasets. As a backbone, Agarwal et al. (2020) chose InceptionV2 with COCO pre-trained weights as a feature extractor, and Sulaiman et al. (2021) selected MatConvnet from MATLAB. On the contrary, Lotter et al. (2021) detected abnormalities with the RetinaNet detection model of ResNet50 backbone with ImageNet weights. In this thesis, we detect masses with the RetinaNet model, but unlike Lotter et al. (2021), we extracted features with three models of backbones: ResNet50, ResNet101, and ResNet152 and compared the results. Each pre-trained model was initialized with ImageNet and COCO weights and trained from scratch.

Two-stage detectors, for instance, R-CNN (Girshick et al., 2014), Faster R-CNN (Girshick, 2015), Feature Pyramid Network (FPN) (Lin et al., 2017), consist of two stages, where the first stage is Region Proposal Network (RPN) to extract regions of objects and the second one is classification to get the object's class and refine the localization. However, two stages are trained separately, making it time- and resource-consuming. On the other hand, RetinaNet is a one-stage detector of two regression and classification subnets, which operate simultaneously. It is recognized that RetinaNet detects objects well with different scales. Due to this, architecture has become valuable and essential in the medical domain. For instance, it helps detect various lesions (Zlocha et al., 2019), (Chen et al., 2022), (Swinburne et al., 2022), (Adachi et al., 2020). RetinaNet consists
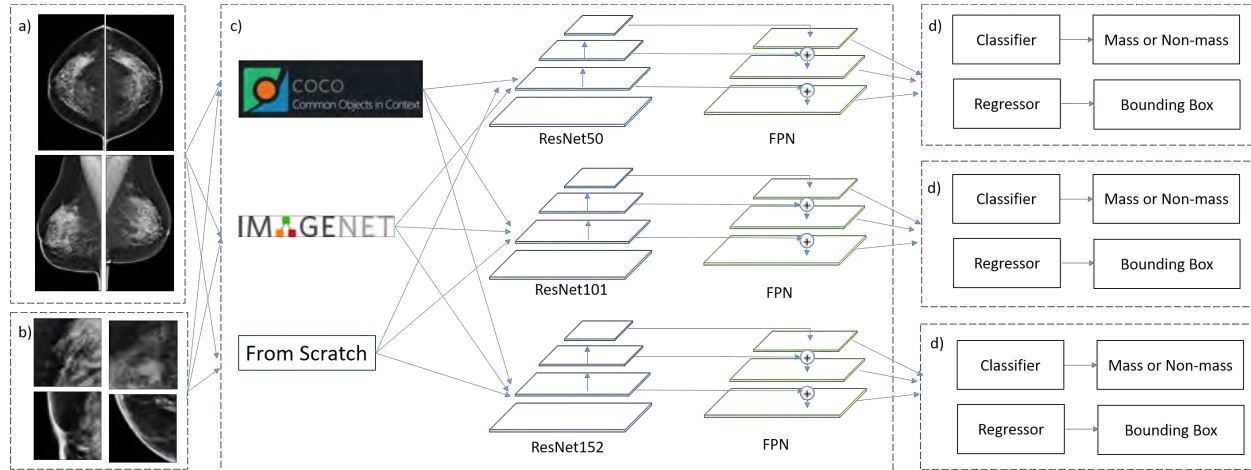
17.6

Figure 5: RetinaNet architecture: a) the first approach with the input of the entire mammogram b) the second approach with the input of patches c) ResNet50, ResNet101, and ResNet152 backbones with ImageNet, COCO weights, and from scratch initializations, and Feature Pyramid Networks, d) two subnetworks of the classification and regression.

of ResNet and FPN backbones. ResNet plays a role as a feature extractor, while FPN creates a multi-scale feature pyramid, scale-invariant, and comprises two subnetworks of classification and regressions. The regression subnetwork predicts bounding boxes, while the classification subnetwork determines the object's class (Lin et al., 2017). In this research we investigate three ResNet50, ResNet101, ResNet152 backbones.

Figure 5 shows the architecture of the detection step, where Figure 5 (a) and (b) perform two approaches with entire mammograms and patches, fed to three backbones with different initialization modes in Figure 5, c (Lin et al., 2017). The weights initialization yields advantages in the starting point, training speed, and higher results. Three pre-trained models of the ResNet extract features are input to the FPN from bottom to up. In each scale, the last layer of the model creates feature maps based on the feature pyramid. FPN is made in the top-down pathway, merged with a respective backbone scale. ResNet models play a role of a feature extractor, while FPN creates a multi-scale feature pyramid, scale-invariant, and comprises two subnetworks of classification and regression (Figure 5, d). The regression subnetwork predicts bounding boxes, while the classification subnetwork determines the object's class (Lin et al., 2017).

The challenging problem in object detection is foreground and background class imbalance. In two-stage detectors, the background classes are narrowed to 1000-2000, while RetinaNet enumerates 100000 predicted objects, spreading densely distributed in the spatial domain and ratios (Lin et al., 2017). The imbalance of negative and positive objects is solved with Focal Loss proposed by Lin et al. (2017), increasing the training efficiency.

ResNet models are deep convolutional networks where the vanishing/exploding gradients problems during backpropagation were solved by adding residual blocks. The layer's output is fed to the next in the traditional architecture. In ResNet, the layer is provided into the next layer and directly into layers located at about 2–3 hops. Also, ResNet50, ResNet101 and ResNet152 have 50, 101 and 152 layers in depth, respectively.

In this study, we train our dataset with the RetinaNet model of ResNet50, ResNet101, and ResNet152 backbones with ImageNet, COCO weights initialization, and from scratch in two approaches with entire mammograms and patches. In this case, the ImageNet and COCO weights act as initialization to speed up the training process, extracted from a large amount of data and transferred to the required small dataset, reaching high effectiveness. Training the model from scratch is a time- and resource-consuming challenge. However, we trained our models from scratch to compare three variants of the initializations.

4.2. Breast mass classification

Several authors apply the first approach of Transfer Learning. The last FC layer is replaced, and three layers of FC, softmax, and classification layers are retrained while all layers are frozen (Yu and Wang, 2019), (Valerio et al., 2019). The standard workflow of Transfer Learning consists of three main steps: to load the model architecture and pre-trained weights on the large ImageNet database with 1000 classes, to replace the last fully-connected layer with a task-specific classification of the dense layer with softmax activation function, and the class number, in our case, benign and malignant,

and the last step is to train the compiled model. This thesis fine-tuned the model, adding L2-regularization, and trained the model with a minimal learning rate. We unfroze the model and added regularization where it was possible to do it to exclude overfitting, enhance the model's generalization, and yield more accurate predictions.

L2-regularization penalizes large weight values and transforms them to close to 0 but not equal. It means that less significant features will have a minor influence over the final prediction while L1-regularization shrinks weights to 0, and features become obsolete. In this case, the L2-norm characterizes as a non-sparse solution with non-zero values and does not act as a feature selector as L1. It helps consider all features to classify masses as benign and malignant. The main difference between L2 and L1 is that L2 penalizes the sum of the square of the weights, while L1 penalizes the sum of the absolute values of the weights.

In the materials section, we investigated that our datasets in the classification step are highly imbalanced. We balanced the class distribution by adding weights to the minority class to overcome this challenge. Class weights compensate limited distribution of the minority class to the majority by adding weight values to the minority class. For instance, we have a 9.9 weight of the benign class, where each sample calculated the loss proportionally to the weights during the training, and the benign class did it with a higher contribution at 9.9 times. This balancing method preserves the algorithm from predicting the prevalent class because of its dominance. If we leave the class distribution in two approaches as it is, in this way, applying the Transfer Learning by replacing the last fully-connected layer with two classes works improperly and learns nothing.

As pre-trained models, we selected four models, ResNet50, InceptionV3, VGG19, and Efficient-NetV2M, to compare the classification results. Following two approaches to the detection step, we trained two datasets with different inputs of the entire mammograms and patches. The predicted bounding boxes were cropped out and pre-processed according to the described steps in the previous sections.

5. Results

This section presents our results regarding all the 18 models trained with two approaches and then classified as benign and malignant. The mass detection method comprises a training RetinaNet model with three backbones and three initialization modes. We chose as backbones ResNet50, ResNet101, and ResNet152, which were trained with ImageNet, COCO weights,

Table 1: Models mAP on the validation subset indicating the best epoch in the first approach.

Models	Epoch	mAP
The first approach		
RetinaNet+ResNet50+ImageNet	40	0.704
RetinaNet+ResNet50+COCO	**17**	**0.709**
RetinaNet+ResNet50+Scratch	57	0.569
RetinaNet+ResNet101+ImageNet	**25**	**0.714**
RetinaNet+ResNet101+COCO	81	0.656
RetinaNet+ResNet101+Scratch	93	0.570
RetinaNet+ResNet152+ImageNet	**21**	**0.715**
RetinaNet+ResNet152+COCO	13	0.000
RetinaNet+ResNet152+Scratch	3	0.000
The second approach		
RetinaNet+ResNet50+ImageNet	14	0.521
RetinaNet+ResNet50+COCO	12	0.519
RetinaNet+ResNet50+Scratch	78	0.515
RetinaNet+ResNet101+ImageNet	**18**	**0.531**
RetinaNet+ResNet101+COCO	**100**	**0.532**
RetinaNet+ResNet101+Scratch	83	0.493
RetinaNet+ResNet152+ImageNet	**13**	**0.531**
RetinaNet+ResNet152+COCO	2	0.000
RetinaNet+ResNet152+Scratch	1	0.000

and from scratch. The weights initialization yields advantages in the starting point, training speed, and higher results. The mAP was utilized as a metric to monitor the training process on the validation subsets, and during the training, we saved a snapshot of each epoch. The total number of epochs in each approach and initialization is 100, and the batch size is 4, with 676 steps in the first approach and 5013 steps in the second approach with Adam optimizer and a 1e-5 learning rate in both approaches. The number of steps depends on the number of images. Due to the dataset size with patches, the number of steps is more significant than the first approach. Additionally, data were augmented with affine random transformations. We trained the same models in two approaches; however, the input differed. As was described in the "Dataset", we extracted whole mammograms and patches of 500×500 pixels and selected them only with masses for training.

The prediction of the bounding boxes of the masses is the same in two approaches. We took each 10th epoch and predicted test datasets with 11 different variants from 10 to 100 epochs, including the best epoch. The best epoch is defined with the highest mAP on the validation subset during the model's training. We forecasted bounding boxes for each epoch and calculated the TPR at the FPPI metric for all models. Moreover, selected models from the detection step are classified by four pre-trained models in terms of

malignancy. The results of the classification step are evaluated with precision, recall, F1-score, and accuracy.

5.1. Breast mass detection results

We fitted the RetinaNet model with three backbones initialized with ImageNet, COCO weights, and from scratch with two approaches. The training process of each model was evaluated by the highest mAP, which could be reached in the epochs. Table 1 shows the performance of mAP for each model, indicating the best epoch in both approaches. As a result, in the first approach, almost all the models showed mAP above 0.7, though in the second approach, mAP is around 0.5. In two approaches, two last RetinaNet models with ResNet152 backbones initialized with COCO weights and from scratch showed the 0-value of mAP and stopped training. According to Table 1, in the first approach, the model RetinaNet of ResNet152 backbone initialized with ImageNet weights demonstrated the maximum 0.71498 mAP and convergence in 21 epochs, while RetinaNet+ResNet101 backbone initialized with the same weights showed the same comparable mAP of 0.71364 at 25 epochs. The fastest convergence was reached by RetinaNet with Resnet50 backbone and COCO weights at 17 epochs and illustrated one of the top mAP of 0.70894 and two models above. Other models demonstrated lower mAP, and it took more epochs and time to perform their top mAP. Despite the convergence at the best epoch, we trained all 100 epochs in each model, saving the snapshot of every epoch.

In the second approach, where the patches are the input to the networks, the best results of mAP are shown by three models of ResNet101 backbone with ImageNet and COCO weights and ResNet152 backbone with the first type of the weights (Table 1). They illustrated more than 0.53 mAP with the convergence before 20 epochs in the first and third models. Moreover, the second model (RetinaNet+ResNet101+COCO) demonstrated the 0.532 mAP at 100 out of 100 epochs. The further training of these variants is not considered in this thesis. We can assume that if we train this model more, it can give some interesting results. In contradistinction to the first approach, the models from scratch in the second approach displayed comparable results with models initialized weights.

Table 2 presents the results of each ten epochs and predictions on the test datasets from 10 to 100 epochs, including the best epochs for two approaches. To compare the results of the detection step, we evaluated our results with TPR at the FPPI metric, where TPR refers to sensitivity and recall. According to the accepted metric, True Positive masses are detected if the IoU is greater than 10%. It means that the predicted

bounding box overlaps its groundtruth by 10%. False Positives are considered if the IoU is less than 10%. We calculated the TPR metric at FPPI for eleven chosen epochs in all approaches' models. Table 2 displays that we achieved 0.938 TPR at 0.84 FPPI in the RetinaNet model with ResNet50 backbone initialized with COCO weights in 50 epochs of the first approach. At the same time, ResNet101 with ImageNet weights demonstrated 0.944 TPR at 0.84 FPPI, which is 0.006 higher than COCO weights. The model with ResNet152 and the same initialization as the abovementioned model demonstrated a top 0.959 TPR at 0.84 FPPI. Furthermore, the ResNet152 backbone with ImageNet weights was the best model with the highest TPR at 0.84 FPPI in the best epoch. Therefore, the RetinaNet with three backbones and weights initialization can be used as the top model in the following classification step of the first approach. In contrast, models from scratch showed the lowest TPR among those initialized by weights, and the models with the highest mAP showed the top TPR parameters. As a result, we determined the predicted masses of three models to classify them with whole mammograms. They are RetinaNet+ResNet50+COCO at 50 epochs, RetinaNet+ResNet101+ImageNet and RetinaNet+ResNet152+ImageNet at the best 25 and 21 epochs, respectively.

Along with the first approach, we compared all training models in the second approach in terms of the metric of TPR at 0.84 FPPI (Table 2). The second approach presented uncompetitive results at ResNet50 backbone from scratch with 0.913 at 0.84 FPPI at 100 epochs and ResNet101 backbone initialized by COCO weights with 0.918 at 0.84 FPPI at 50 epochs. At the same time, the ResNet50 backbone with ImageNet initialization performed 0.905 TPR at 0.84. The top 0.918 TPR at 0.84 FPPI in the second approach was established at 50 epochs, while in the first approach, it was 21 epochs. As a result, according to the TPR at 0.84 FPPI, we selected two models, RetinaNet of ResNet50 backbone from scratch at 100 epochs and ResNet101 with COCO weights initialization at 50 epochs to the classification step.

Additionally, we plotted the FROC curves from 10 to 100 epochs with a step of 10 epochs with the best epochs. The best epoch was chosen with the best mAP of the validation dataset during the training. The FROC curve shows the relationship between TPR on the y-axis and FPPI on the x-axis. We plotted the FROC of seven models and epochs with a range of confidence degrees of two approaches (Figure 6 (a-g) and Figure 7 (a-g)). Figures 6 (h) and 7 (h) perform the FROC with the chosen epochs of top TPR in two approaches to visually compare curves among the models.

Table 2: Performance comparison of mass detection models of the first approach in epochs with step 10. The metrics correspond to the True Positive Rate (TPR) at 0.84 False Positive Per Image.

Model	TPR at 0.84 FPPI										
Epoch	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**	**BEST**
The first approach											
RetinaNet+ ResNet50+ImageNet	0.915	0.921	0.934	0.932	0.920	0.912	0.918	0.914	0.908	0.920	0.932
RetinaNet+ ResNet50+COCO	0.912	0.919	0.923	0.923	**0.938**	0.929	0.926	0.922	0.929	0.910	0.923
RetinaNet+ ResNet50+Scratch	0.776	0.884	0.831	0.854	0.870	0.865	0.866	0.867	0.867	0.868	0.867
RetinaNet+ ResNet101+ImageNet	0.938	0.934	0.931	0.935	0.922	0.902	0.916	0.905	0.894	0.900	**0.944**
RetinaNet+ ResNet101+COCO	0.790	0.812	0.852	0.867	0.861	0.871	0.879	0.883	0.890	0.887	0.883
RetinaNet+ ResNet101+Scratch	0.847	0.867	0.866	0.890	0.838	0.846	0.825	0.850	0.882	0.873	0.855
RetinaNet+ ResNet152+ImageNet	0.936	0.934	0.930	0.897	0.901	0.908	0.931	0.896	0.914	0.891	**0.959**
The second approach											
RetinaNet+ ResNet50+ImageNet	0.866	0.865	0.870	0.887	0.890	0.897	0.858	0.815	0.771	0.776	0.905
RetinaNet+ ResNet50+COCO	0.878	0.896	0.892	0.873	0.877	0.882	0.872	0.857	0.851	0.855	0.890
RetinaNet+ ResNet50+Scratch	0.847	0.844	0.836	0.842	0.884	0.881	0.891	0.892	0.899	**0.913**	0.910
RetinaNet+ ResNet101+ImageNet	0.844	0.897	0.860	0.892	0.879	0.839	0.833	0.812	0.878	0.859	0.893
RetinaNet+ ResNet101+COCO	0.857	0.853	0.906	0.908	**0.918**	0.904	0.899	0.910	0.918	0.917	0.917
RetinaNet+ ResNet101+Scratch	0.792	0.808	0.844	0.875	0.873	0.875	0.870	0.874	0.880	0.911	0.904
RetinaNet+ ResNet152+ImageNet	0.890	0.877	0.839	0.857	0.838	0.849	0.799	0.843	0.791	0.795	0.824

5.2. Breast mass classification results

The obtained results of the detection step are used in the classification task. We received predicted masses in two approaches from the whole mammograms and patches and selected 3 models in RetinaNet of ResNet50 backbone with COCO weights at 50 epochs, ResNet101 and ResNet152 backbones with ImageNet weights initialization at the best 25 and 21 epochs, respectively, in the first approach. In the second approach, we opted for two models of ResNet50 backbone from scratch and ResNet101 with COCO weights initializations at 100 and 50 epochs. The original test subsets, extracted from the original OMI-DB dataset and predicted in the detection step, consist of 776 and 5767 masses in two approaches, respectively. Table 3 shows the distribution of benign and malignant classes in the original and predicted test subsets. It is seen that subsets are imbalanced with the majority of the malignant class.

Moreover, due to the different number of the predicted masses in the detection step, the size of the test subsets varies. All models were chosen with 0.5 of the confidence degree of the predicted masses. Table 3 presents the results of the detection step, where in the first approach, RetinaNet+ResNet50+COCO predicted 54 benign masses out of 82 of the original test subset and 583 malignant out of 694. The last model predicted only 70.3% of benign and 67.1% of malignant from the original masses. In the second approach, only 270 benign masses out of 670 and 2361 out of 5097 malignant were predicted in the RetinaNet+ResNet50+Scratch model, while the next model forecasted 266 and 2440 benign and malignant masses respectively. However, only predicted masses were classified, while unpredicted masses were discarded.

We implemented fine-tuning Transfer Learning by unfreezing all layers, adding L2-penalty, and retraining
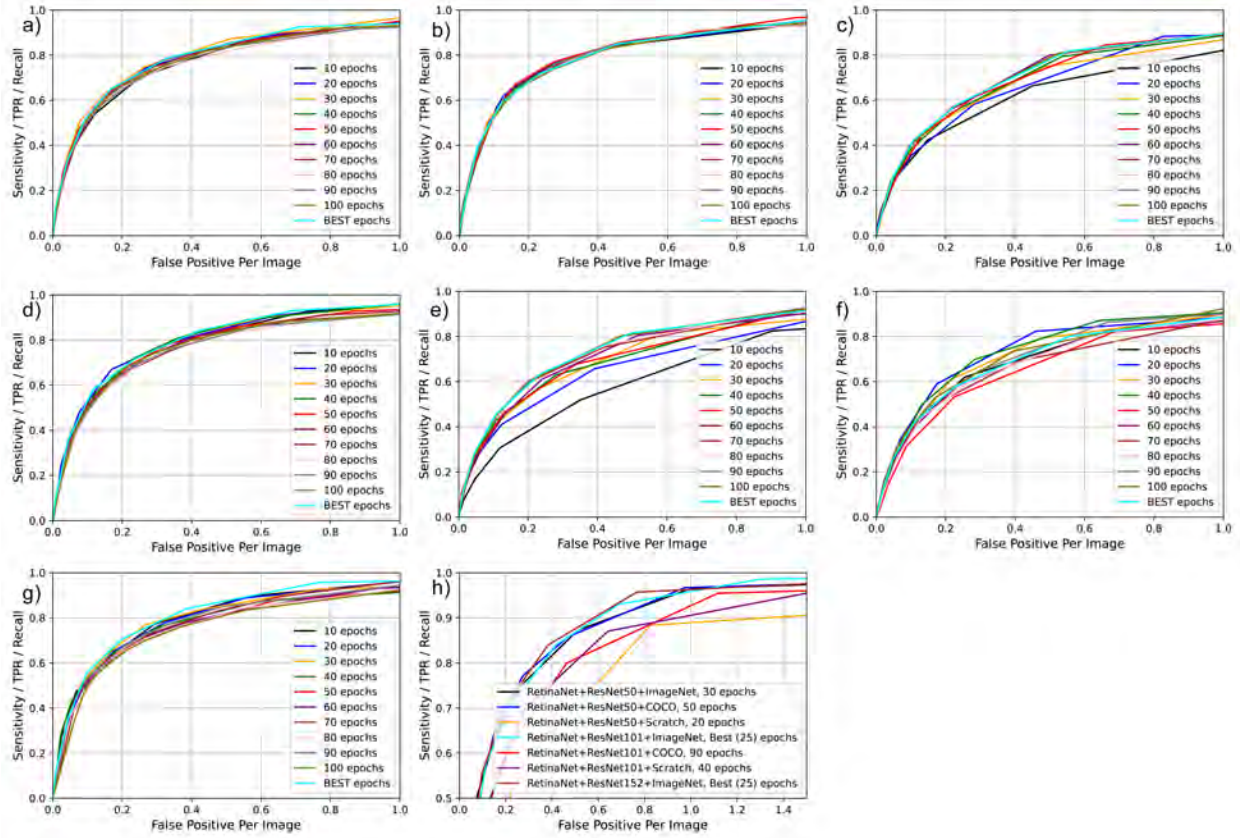
17.10

Figure 6: Free-response Receiver Operating Characteristic curve of RetinaNet model with three types of the initialization in the first approach: a) ResNet50 with ImageNet weights, b) ResNet50 with COCO weights, c) ResNet50 from scratch, d) ResNet101 with ImageNet weights, e) ResNet101 with COCO weights, f) ResNet101 from scratch, g) ResNet152 with ImageNet weights, h) FROC curves of all models in the first approach with the highest TPR parameter

Table 3: Class distribution of the selected models of the first and second approaches

Model	Benign	Malignant
The first approach		
Original Test subset	82	694
RetinaNet+ResNet50+ COCO at 50 epochs	54	583
RetinaNet+ResNet101+ ImageNet at BEST (25) epochs	54	632
RetinaNet+ResNet152+ ImageNet at BEST (21) epochs	38	466
The second approach		
Original Test subset	670	5097
RetinaNet+ResNet50+ Scratch at 100 epochs	270	2361
RetinaNet+ResNet101+ COCO at 50 epochs	266	2440

the model with class weights using four pre-trained models. The algorithm was described in the "Methodology" section. We evaluated the results with precision, recall, F1-score, and accuracy (Table 4). The highest precision, 0.95, was conducted with the ResNet50 and EfficientNetV2M pre-trained model, which means that 95% of masses were classified correctly. Simultaneously, the precision in the VGG19 model is a maximum of 0.93. The precision in all datasets and pre-trained models was more than 0.92.

The recall metric, also known as sensitivity, ranges between 0.61 and 0.98. The recall metric shows how many samples are identified with the correct class according to the total number of images. The highest 0.98 recall of RetinaNet with ResNet152 backbone with ImageNet demonstrates that 98% are predicted correctly in VGG19 and EfficientNetV2M models. Models of the second approach RetinaNet+ResNet50+Scratch and RetinaNet+ResNet101+COCO showed the highest 0.90 and 0.93 recall in ResNet50 and EfficientNetV2M respectively. The maximum 0.96 F1-score showed RetinaNet with ResNet152 backbone with ImageNet weights trained with the ResNet50, VGG19, and
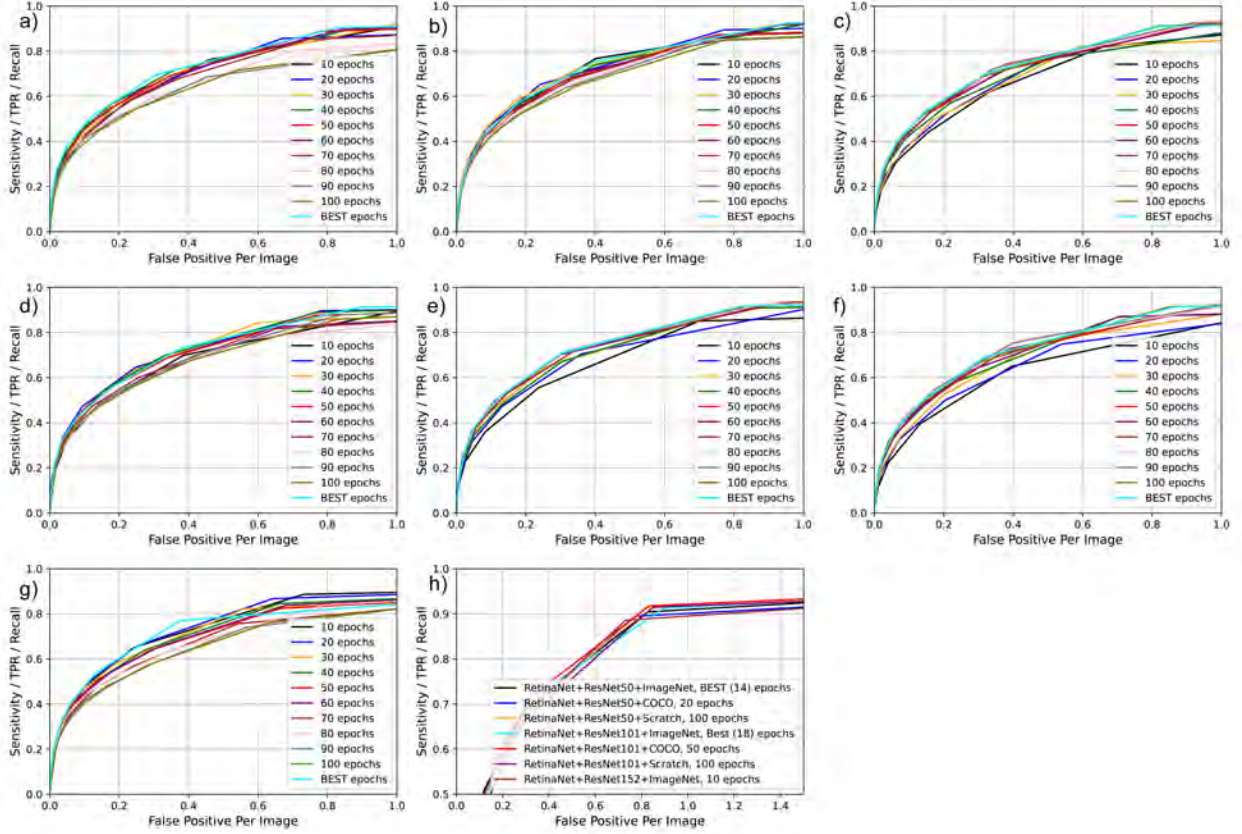
Figure 7: Free-response Receiver Operating Characteristic curve of RetinaNet model with three backbones and initializations in the second approach: a) ResNet50 with ImageNet weights, b) ResNet50 with COCO weights, c) ResNet50 from scratch, d) ResNet101 with ImageNet weights, e) ResNet101 with COCO weights, f) ResNet101 from scratch, g) ResNet152 with ImageNet weights, h) FROC curves of all models in the second approach with the highest TPR parameter

EfficientNetV2M models. F1-score is the combination of precision and recall, which is helpful in the imbalanced datasets across benign and malignant samples, balancing the overall score based on them.

According to the results in Table 4, we conclude that the best results were obtained with three pre-trained models, such as ResNet50, VGG19, and EfficientNetV2M, with the dataset of RetinaNet+ResNet152+ImageNet in terms of the precision, recall, F1-score, and accuracy. Two models of the second approach from the detection step (RetinaNet+ResNet50+Scratch and RetinaNet+ResNet101+COCO) presented the lowest recall, F1-score, and accuracy values. As a result, we selected the EfficientNetV2M model of the RetinaNet+ResNet152+ImageNet as the best performance in all metrics in the classification step.

5.3. Qualitative results

Figure 8 illustrates the qualitative results of the predicted bounding boxes of masses on the entire mammograms of the OMI-H dataset of two approaches in the detection task. They performed examples from all

the five models with three types of the predicted bounding boxes: predictions with high confidence (80%-90% and higher) (Figure 8, a), predicted wrong (Figure 8, b), and undetected masses (Figure 8, c). We take the same color parameters of the bounding boxes visualization as Agarwal et al. (2020): green bounding boxes are groundtruths, yellow – True Positives, and red – False Positives. The confidence score is displayed above True Positive predictions.

6. Discussion

Agarwal et al. (2020) utilized two subsets of OMI-DB with Hologic and General Electric manufacturers. As a detection model, they utilized Faster R-CNN with InceptionV2 backbone as a feature extractor with COCO weights achieving 0.87 TPR at 0.84 FPPI. On the other hand, Sulaiman et al. (2021) and Lotter et al. (2021) developed two tasks of mass detection and classification. Sulaiman et al. (2021) applied Faster R-CNN as Agarwal et al. (2020), though to the combination of the MIAS and CBIS-DDSM datasets with VGG19, InceptionV3, and MatConvNet to extract features. In this study, we utilized RetinaNet detection model

Table 4: Classification step evaluation of the two detection approaches: precision, recall, F1-score, and accuracy

Model	ResNet50	InceptionV3	VGG19	EfficientNetV2M
Precision				
RetinaNet+ResNet50+COCO	0.93	0.93	0.93	0.93
RetinaNet+ResNet101+ImageNet	0.94	0.93	0.93	0.94
RetinaNet+ResNet152+ImageNet	**0.95**	0.93	**0.93**	**0.95**
RetinaNet+ResNet50+Scratch	0.92	0.93	0.93	0.95
RetinaNet+ResNet101+COCO	0.93	0.93	0.92	0.93
Recall				
RetinaNet+ResNet50+COCO	0.95	0.92	0.97	0.96
RetinaNet+ResNet101+ImageNet	0.95	0.90	0.98	0.95
RetinaNet+ResNet152+ImageNet	**0.97**	0.92	**0.98**	**0.98**
RetinaNet+ResNet50+Scratch	0.90	0.80	0.88	0.61
RetinaNet+ResNet101+COCO	0.90	0.81	0.88	0.93
F1-score				
RetinaNet+ResNet50+COCO	0.94	0.92	0.95	0.95
RetinaNet+ResNet101+ImageNet	0.94	0.91	0.95	0.94
RetinaNet+ResNet152+ImageNet	**0.96**	0.93	**0.96**	**0.96**
RetinaNet+ResNet50+Scratch	0.91	0.86	0.90	0.74
RetinaNet+ResNet101+COCO	0.92	0.86	0.90	0.93
Accuracy				
RetinaNet+ResNet50+COCO	0.89	0.86	0.90	0.90
RetinaNet+ResNet101+ImageNet	0.89	0.84	0.91	0.90
RetinaNet+ResNet152+ImageNet	**0.92**	0.87	**0.92**	**0.93**
RetinaNet+ResNet50+Scratch	0.84	0.77	0.83	0.62
RetinaNet+ResNet101+COCO	0.85	0.77	0.83	0.88

as Lotter et al. (2021), however with three different backbones of ResNet50, ResNet101 and ResNet152, while Lotter et al. (2021) used only ResNet50. Moreover, unlike Lotter et al. (2021) which initialized the detection model with ImageNet weights, we obtained results with ImageNet and COCO weights and from scratch. Considering the FFDMs size, only Lotter et al. (2021) extracted patches from the mammograms, while other authors fed entire images to the models. Compared with state-of-the-art papers in the detection, we implemented two approaches, entire mammograms, and patches.

Agarwal et al. (2020), Lotter et al. (2021), and our thesis used the same OMI-DB dataset. However, the cases with images were not identical due to different subsets of the OMI-DB being provided. On the one hand, our investigation detected masses in the images produced by Hologic manufacture. However, Agarwal et al. (2020), on a par with Hologic processed images of Siemens, General Electric, and Philips manufacturers, unlike Lotter et al. (2021) with Hologic and General Electric images. At the same time, the distribution of benign and malignant images in OMI-H of Agarwal et al. (2020) paper seems almost similar. Agarwal et al. (2020) extracted 2042 malignant and 842 benign images. In our case, we selected 3524 images with masses, 3162 malignant and 362 benign, which

is 90% versus 10% in the percentage ratio in our dataset.

According to Table 5, many papers used INbreast, mini-MIAS, and DDSM datasets to implement mass detection methods. For instance, Kozegar et al. (2013) reached 0.87 TPR at 3.67 FPPI and 0.91 TPR at 4.8 FPPI in two datasets. Akselrod et al. (2017) and Shen et al. (2020) utilized private datasets in line with the INbreast dataset, where they performed 0.93 TPR at 0.56 FPPI and 0.879 TPR at 0.5 FPPI, respectively. Many papers applied their method to the public datasets and their merge, which are available with free access to obtain larger datasets, fine-tune models, and implement more robust methods. Not many articles use the OMI-DB dataset to detect mammogram abnormalities with different methods. We implemented our pipeline using the Transfer Learning method with the RetinaNet model as a baseline and different backbones initiated by ImageNet, COCO weights, and from scratch in two approaches.

This research aimed to investigate how a Transfer Learning approach affects the performance of a system for breast cancer detection. In particular, we considered a detector based on RetinaNet with two different Transfer Learning settings (ImageNet and COCO) and trained from scratch. The top model RetinaNet of ResNet152 backbone with ImageNet weights showed
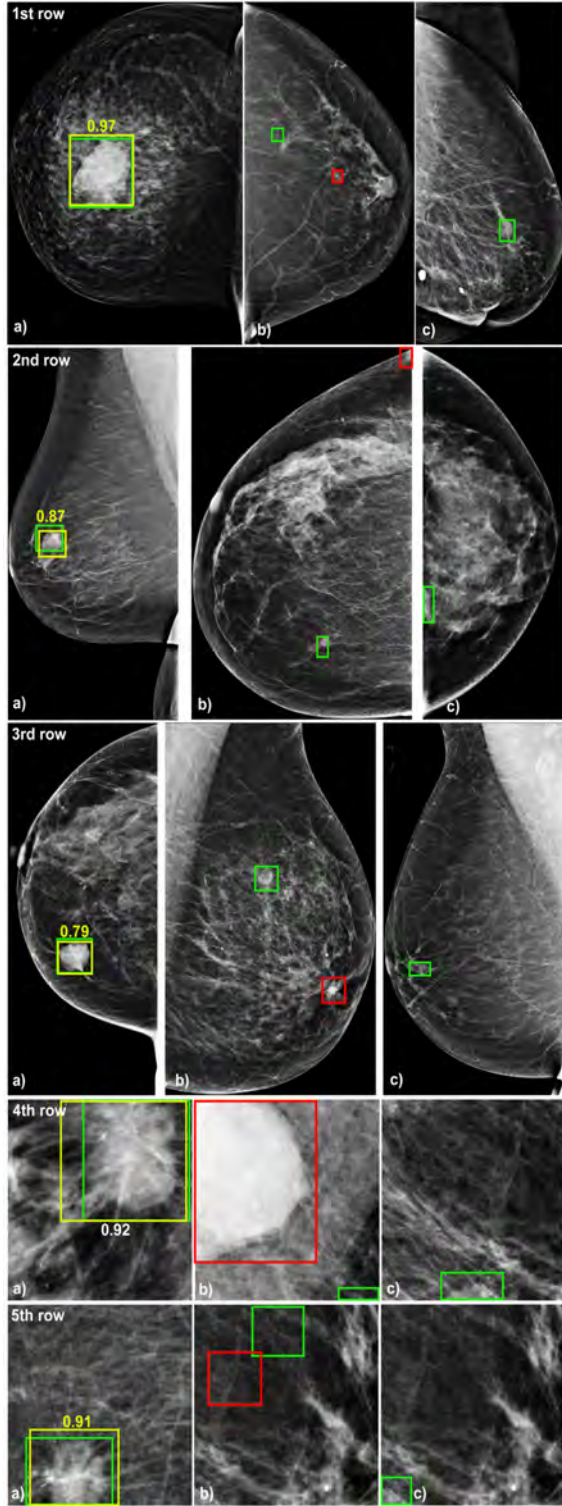
Figure 8: Qualitative results of the mass detection in OMI-H dataset, the 1st row demonstrates the results of RetinaNet+ResNet50+COCO, the 2nd row RetinaNet+ResNet101+ImageNet, the 3rd row RetinaNet+ResNet152+ImageNet, the 4th row RetinaNet+ResNet101+COCO, the 5th row RetinaNet+ResNet101+Scratch: a – True Positive detections with high objectness score; b – False Positive detections; c – undetected masses. The numbers shown in the images correspond to the confidence of being mass. The color of bounding boxes: green – groundtruths, yellow – True Positives, red – False Positives.

Table 5: Comparison of the mass detection between the proposed framework and the published results

Method	TPR at FPPI	Dataset
Kozegar et al. (2013)	0.87 at 3.67	INbreast
	0.91 at 4.8	mini-MIAS
Akselrod-Ballin et al. (2017)	0.93 at 0.56	INbreast
	0.90 at 1.0	Private
Shen et al. (2020)	0.879 at 0.5	INbreast
	0.948 at 2.0	Private
Anitha et al. (2017)	0.935 at 0.62	mini-MIAS
	0.925 at 1.06	DDSM
Te Brake et al. (2000)	0.55 at 0.10	DDSM
Dhungel et al. (2017)	0.90 at 1.30	INbreast
Ribli et al. (2018)	0.90 at 0.30	INbreast
Jung et al. (2018)	0.94 at 1.30	INbreast
Agarwal et al. (2019)	0.98 at 1.67	INbreast
Agarwal et al. (2020)	0.87 at 0.84	OMI-H
Proposed framework:		
RetinaNet+ ResNet50+COCO	0.938 at 0.84	OMI-H
RetinaNet+ ResNet101+ImageNet	0.944 at 0.84	OMI-H
RetinaNet+ ResNet152+ImageNet	0.959 at 0.84	OMI-H

0.959 TPR at 0.84 FPPI, while Agarwal et al. (2020) reached 0.87 TPR at 0.84 FPPI (Table 5). Table 5 compares the papers on mass detection and reached results in different datasets, and we proposed our framework with three backbones and two types of initialization.

Furthermore, predicted masses in the detection step were classified as benign and malignant. Minority of papers implement detection and classification tasks simultaneously. Mainly they are developed separately or considered for future work in many cases. Most papers of the classification task performed in Table 6 used the public datasets in the detection step for classifying masses into benign and malignant, pathological and non-pathological and etc., while scarcity of papers with OMI-H classifying masses is observed. This thesis presented our results using Transfer Learning fine-tuning approach by adding L2-regularization, which gave us

Table 6: Comparison of the classification between proposed framework and the published results

Method	Model	Acc.	Dataset
Valerio et al. (2019)	Inception-ResNet-v2	94.34%	MAMMO-SET
Falconi et al. (2020)	VGG19	90.9%	INbrest
Yu and Wang (2019)	ResNet18	95.91%	mini-MIAS
Alruwaili and Gouda (2022)	ResNet50	89.5%	MIAS
Khamparia et al. (2021)	VGG16	86.9%	MIAS
Mahmood et al. (2021)	VGG19+SVM	97.8%	MIAS, INBreast, Private
Altaf (2021)	PCNN+GoogLeNet	98.72%	DDMS
Proposed framework:			
RetinaNet+	ResNet50	92%	OMI-H
ResNet152+	VGG19	92%	OMI-H
ImageNet	Efficient-NetV2M	93%	OMI-H

92%-93% accuracy with ResNet50, VGG19 and EfficientNetV2M. However, we faced the problem of the highly imbalanced distribution of two classes in the datasets. These results were obtained using the bounding boxes predicted from the whole mammograms from the detection step.

7. Conclusions

In conclusion, the mass detection task was efficiently investigated using Transfer Learning with different learning settings. We compared three initialization modes in three backbones, where the model with weights showed state-of-the-art results while models from scratch demonstrated lower results. Moreover, we compared two approaches where entire mammograms and extracted patches were used as input to the models, where the first approach reached the highest TPR values.

The classification pipeline was also implemented using pre-trained models of Transfer Learning. However, the challenge was to train the model classifying a highly-imbalanced dataset, where we found a solution to fine-tune the model by adding regularization and class weights to the minority class. The results showed that the entire mammogram approach could efficiently predict masses with bounding boxes and classify them as benign or malignant.

References

Adachi, M., Fujioka, T., Mori, M., Kubota, K., Kikuchi, Y., Xiaotong, W., Oyama, J., Kimura, K., Oda, G., Nakagawa, T., Uetake, H., Tateishi, U., 2020. Detection and diagnosis of breast cancer using artificial intelligence based assessment of maximum intensity projection dynamic contrast-enhanced magnetic resonance images. Diagnostics (Basel) 10, 330.

Agarwal, R., Diaz, O., Lladó, X., Yap, M.H., Martí, R., 2019. Automatic mass detection in mammograms using deep convolutional neural networks. J. Med. Imaging (Bellingham) 6, 1.

Agarwal, R., Díaz, O., Yap, M.H., Lladó, X., Martí, R., 2020. Deep learning for mass detection in full field digital mammograms. Comput. Biol. Med. 121, 103774.

Akselrod-Ballin, A., Karlinsky, L., Hazan, A., Bakalo, R., Horesh, A.B., Shoshan, Y., Barkan, E., 2017. Deep learning for automatic detection of abnormal findings in breast mammography, in: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer International Publishing, Cham, pp. 321–329.

Alruwaili, M., Gouda, W., 2022. Automated breast cancer detection models based on transfer learning. Sensors (Basel) 22, 876.

Altaf, M.M., 2021. A hybrid deep learning model for breast cancer diagnosis based on transfer learning and pulse-coupled neural networks. Math. Biosci. Eng. 18, 5029–5046.

Anitha, J., Peter, J.D., Pandian, S.I.A., 2017. A dual stage adaptive thresholding (DuSAT) for automatic mass detection in mammograms, comput. Comput," Comput. Methods Programs Biomed 138, 93–104.

Battisti, F., Falini, P., Gorini, G., Bianchi, P., Armaroli, P., Giubilato, P., Giorgi Rossi, P., Zorzi, M., Battagello, J., Senore, C., Zappa, M., Mantellini, P., 2022. Cancer screening programmes in italy during the COVID-19 pandemic: an update of a nationwide survey on activity volumes and delayed diagnoses.: Cancer screening and covid-19 pandemic. Annali Dell'Istituto Superiore Di Sanità 58, 16–24.

Bria, A., Marrocco, C., Borges, L.R., Molinara, M., Marchesi, A., Mordang, J.J., Karssemeijer, N., Tortorella, F., 2018. Improving the automated detection of calcifications using adaptive variance stabilization. IEEE Trans. Med. Imaging 37, 1857–1864.

Bria, A., Marrocco, C., Karssemeijer, N., Molinara, M., Tortorella, F., 2016. Deep cascade classifiers to detect clusters of microcalcifications, in: Breast Imaging. Springer International Publishing, Cham, pp. 415–422.

Chen, J., Li, P., Xu, T., Xue, H., Wang, X., Li, Y., Lin, H., Liu, P., Dong, B., Sun, P., 2022. Detection of cervical lesions in colposcopic images based on the RetinaNet method. Biomed. Signal Process. Control 75, 103589.

Deng, J., Berg, A.C., Li, K., Fei-Fei, L., 2010. What does classifying more than 10,000 image categories tell us?, in: Computer Vision – ECCV 2010. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 71–84.

Dhungel, N., Carneiro, G., Bradley, A.P., 2017. A deep learning approach for the analysis of masses in mammograms with minimal user intervention, med. Image Anal 37, 114–128.

Elia, C., Marrocco, C., Molinara, M., Tortorella, F., 2008. Detection of clusters of microcalcifications in mammograms: A multi classifier approach, in: 21st IEEE International Symposium on Computer-Based Medical Systems.

Falconi, L., Perez, M., Aguilar, W., Conci, A., 2020. Transfer learning and fine tuning in mammogram BI-RADS classification, in: 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS), IEEE.

Gathani, T., Reeves, G., Dodwell, D., Horgan, K., Kearins, O., Kan, S.W., Sweetland, S., 2022. Impact of the COVID-19 pandemic on breast cancer referrals and diagnoses in 2020 and 2021: a population-based study in england. Br. J. Surg. 109, e29–e30.

Girshick, R., 2015. Fast R-CNN, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE.

Halling-Brown, M.D., Warren, L.M., Ward, D., Lewis, E., Mackenzie, A., Wallis, M.G., Wilkinson, L.S., Given-Wilson, R.M., McAvinchey, R., Young, K.C., 2021. OPTIMAM mammography image database: A large-scale resource of mammography images and clinical data. Radiol Artif Intell 3, e200103.

Jung, H., Kim, B., Lee, I., Yoo, M., Lee, J., Ham, S., Woo, O., Kang, J., 2018. Detection of masses in mammograms using a one-stage object detector based on a deep convolutional neural network. PLoS One 13, e0203355.

Khamparia, A., Bharati, S., Podder, P., Gupta, D., Khanna, A., Phung, T.K., Thanh, D.N.H., 2021. Diagnosis of breast cancer based on modern mammography using hybrid transfer learning. Multidimens. Syst. Signal Process. 32, 747–765.

Kozegar, E., Soryani, M., Minaei, B., Domingues, I., 2013. Assessment of a novel mass detection algorithm in mammograms. J. Cancer Res. Ther. 9, 592–600.

Le Bihan Benjamin, C., Simonnet, J.A., Rocchi, M., Khati, I., Ménard, E., Houas-Bernat, E., Méric, J.B., Bousquet, P.J., 2022. Monitoring the impact of COVID-19 in france on cancer care: a differentiated impact. Sci. Rep. 12, 4207.

Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft COCO: Common objects in context, in: Computer Vision – ECCV 2014. Springer International Publishing, Cham, pp. 740–755.

Lotter, W., Diab, A.R., Haslam, B., Kim, J.G., Grisot, G., Wu, E., Wu, K., Onieva, J.O., Boyer, Y., Boxerman, J.L., Wang, M., Bandler, M., Vijayaraghavan, G.R., Gregory Sorensen, A., 2021. Robust breast cancer detection in mammography and digital breast tomosynthesis using an annotation-efficient deep learning approach. Nat. Med. 27, 244–249.

Mahmood, T., Li, J., Pei, Y., Akhtar, F., 2021. An automated in-depth feature learning algorithm for breast abnormality prognosis and robust characterization from mammography images using deep transfer learning. Biology (Basel) 10, 859.

Mammography, M., Heath, K., Bowyer, D., Kopans, R., Moore, W.P., 2001. Proceedings of the Fifth International Workshop on Digital Mammography, M.J. Yaffe. Medical Physics Publishing.

Marchesi, A., Bria, A., Marrocco, C., Molinara, M., Mordang, J.J.,

Tortorella, F., Karssemeijer, N., 2017. The effect of mammogram preprocessing on microcalcification detection with convolutional neural networks, in: 2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS), IEEE.

Marrocco, C., Molinara, M., Tortorella, F., 2005. Algorithms for detecting clusters of microcalcifications in mammograms, in: Image Analysis and Processing – ICIAP 2005. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 884–891.

Monticciolo, D.L., Malak, S.F., Friedewald, S.M., Eby, P.R., Newell, M.S., Moy, L., Destounis, S., Leung, J.W.T., Hendrick, R.E., Smetherman, D., 2021. Breast cancer screening recommendations inclusive of all women at average risk: Update from the ACR and society of breast imaging. J. Am. Coll. Radiol. 18, 1280–1288.

Moreira, I.C., Amaral, I., Domingues, I., Cardoso, A., Cardoso, M.J., Cardoso, J.S., 2012. INbreast: toward a full-field digital mammographic database. Acad. Radiol. 19, 236–248.

Ribes, Pareja, Sanz, Mosteiro, Jm, E., Esteban, Gálvez, Osca, Rodenas, Pérez-Sust, P., Jm, B., 2022. Cancer diagnosis in catalonia (spain) after two years of COVID-19 pandemic: an incomplete recovery. ESMO Open , 100486.

Ribli, D., Horváth, A., Unger, Z., Pollner, P., Csabai, I., 2018. Detecting and classifying lesions in mammograms with deep learning, sci. Sci," Sci. Rep 8.

Ruiz-Medina, S., Gil, S., Jimenez, B., Rodriguez-Brazzarola, P., Diaz-Redondo, T., Cazorla, M., Muñoz-Ayllon, M., Ramos, I., Reyna, C., Bermejo, M.J., Godoy, A., Torres, E., Cobo, M., Galvez, L., Rueda, A., Alba, E., Ribelles, N., 2021. Significant decrease in annual cancer diagnoses in spain during the COVID-19 pandemic: A real-data study. Cancers (Basel) 13, 3215.

Savelli, B., Bria, A., Molinara, M., Marrocco, C., Tortorella, F., 2020. A multi-context CNN ensemble for small lesion detection, " artif. Artif. Artif. Intell. Med .

Sawyer-Lee, R., Gimenez, F., Hoogi, A., Rubin, D., 2016. Curated breast imaging subset of DDSM.

Sechopoulos, I., Teuwen, J., Mann, R., 2021. Artificial intelligence for breast cancer detection in mammography and digital breast tomosynthesis: State of the art. Semin. Cancer Biol. 72, 214–225.

Shen, R., Yao, J., Yan, K., Tian, K., Jiang, C., Zhou, K., 2020. Unsupervised domain adaptation with adversarial learning for mass detection in mammogram, neurocomputing. Neurocomputing .

Suckling, J., Parker, J., Dance, D., Astley, S., Hutt, I., Boggis, C., Ricketts, I., Stamatakis, E., Cerneaz, N., Kok, S., Taylor, P., Betal, D., Savage, J., 2015. Mammographic image analysis society (MIAS) database v1.21.

Sulaiman, S.N., Hassan, N.A., Isa, I.S., Abdullah, M.F., Soh, Z.H.C., Jusman, Y., 2021. Mass detection in digital mammogram image using convolutional neural network (CNN), in: 2021 11th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), IEEE.

Swinburne, N.C., Yadav, V., Kim, J., Choi, Y.R., Gutman, D.C., Yang, J.T., Moss, N., Stone, J., Tisnado, J., Hatzoglou, V., Haque, S.S., Karimi, S., Lyo, J., Juluru, K., Pichotta, K., Gao, J., Shah, S.P., Holodny, A.I., Young, R.J., MSK MIND Consortium, 2022. Semisupervised training of a brain MRI tumor detection model using mined annotations. Radiology 303, 80–89.

Te Brake, G.M., Karssemeijer, N., Hendriks, J.H.C.L., 2000. An automatic method to discriminate malignant masses from normal tissue in digital mammograms1. Phys. Med. Biol 45, 2843–2857.

Valerio, L.M., Alves, D.H.A., Cruz, L.F., Bugatti, P.H., de Oliveira, C., Saito, P.T.M., 2019. DeepMammo: Deep transfer learning for lesion classification of mammographic images, in: 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS), IEEE.

Yu, X., Wang, S.H., 2019. Abnormality diagnosis in mammograms by transfer learning based on ResNet18. Fundam. Inform. 168, 219–230.

Zlocha, M., Dou, Q., Glocker, B., 2019. Improving RetinaNet for CT lesion detection with dense masks from weak RECIST labels, in: Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 402–410.
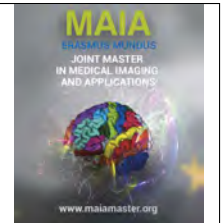
Classification of malignant nodules from 2D ultrasound thyroid images using Deep Convolutional Neural Networks

Tewele Weletnsea Tareke, Alain Lalande (PhD), Sarah Leclerc (PhD)

ImViA Laboratory, Université Bourgogne Franche-Comté, Dijon, France

Abstract

Thyroid nodule is a type of disease that affects the thyroid gland, a small gland at the base of the neck that produces hormones. In clinical routine, thyroid nodules are usually detected manually by expert physician or radiologists. Manual classification of nodules by physician has several drawbacks: it is time consuming, inaccurate and tends to expose patients to unnecessary fine needle aspirations (FNA). It also suffers from inter-and intra-observer variabilities. Manual classification also is time consuming, less accurate and exposes to unnecessary fine needle aspiration (FNA) biopsies, which brings a lot of stress to patients. Thus, it is of considerable interest to develop an automatic and accurate thyroid nodule classification system. However, automatic methods struggle in presence of noise, artifact and low contrast, all characteristics of ultrasound imaging. In this paper, we propose an automatic computer Aided Diagnosis System CAD for the classification of thyroid nodules using a fine-tuned Deep Learning model based on Densenet121 architecture in which an attention module is incorporated (Densenet-Attention). This CAD was developed using 595 thyroid nodule images that were fully annotated based on the Bethesda scores established from the biopsy. Out of these samples, 252 images were annotated as positive and 342 were annotated as negative. 51 images are used as test set to validate our proposed method. Several image enhancement methods were applied, such as histogram equalization, artifacts removal, and range scaling. We augmented the dataset with synthetic images obtained with label-preserving transformations and added a convolutional block appended with an attention module to extract global feature maps and forward them as inputs to the decision layers. Two attention modules (Channel and Spatial) were integrated in this proposed architecture, aiming to help the network focus on the most important feature maps and locations. Our method used a focal loss to encourage prediction accuracy by penalizing the misclassified examples. Moreover, we demonstrated explainability of the decision using Gradient-weighted class activation maps(Grad-cam) to identify the most substantial region of the images. The proposed method is evaluated on datasets acquired from Hospitals in Bastia and Dijon. On the test set, our best approach achieved an average accuracy of **90.70%**, F1-score of **92.16%** and sensitivity of **96.42%**, which compares favorably to the state-of-the-art. The proposed method also outperformed similar methods and demonstrated that integrating attention modules improves the classification result.

Keywords: Thyroid nodule, EU-TIRADS, Ultrasound Image, Interoperability, Classification, Deep Learning,

1. Introduction

Thyroid nodules are irregular overgrowth of tissues in the thyroid gland. Most thyroid nodules are not consequential and do not cause symptoms. Some people have one nodule, while others have many. According to the American Cancer Society, there are still numerous deaths due to malignant thyroid nodules. While a moderate percentage of thyroid nodules are cancerous (between 3 and 7% (Hambly et al., 2011)), the death rate for thyroid cancer tends to increase: it augmented by 0.6% per year between 2009 to 2018 according to the Key Statistics for Thyroid Cancer. Autopsy studies have reported incidental thyroid nodules subjectivity up to 50%.

In consequence, early detection and treatment of malignant nodules are very significant. Ultrasound (US) imaging techniques have become an important diagnostic tool in the assessment of thyroid nodules. Though

US images are faster to acquire and effective to analyse thyroid nodules, computed tomography(CT) and Magnetic resonance imaging(MRI) can also be used as imaging tools (Peng et al., 2017). Thyroid ultrasonography has the advantages of being a noninvasive, low-priced procedure widely used to detect and evaluate thyroid nodules risk of being malignant. It plays an important role in providing information such as the nodule positions, dimensions, orientation, and pathologic changes. All in all, it is a highly tactful and core modality for the detection of malignant nodules, though its diagnostic value varies from study to study. Identification of malignancy level is dependent on the quality of the exam, that in turn depends on the physician. Therefore, inter-observer variability exists for the assessment of thyroid nodules. The experience of the sonographer to properly acquire and label the image is substantial, because an inaccurate US capture of a nodule might result in unnecessary fine-needle aspiration (biopsy). Hence, an accurate automated diagnosis system is required to avoid unnecessary punctures.

Table 1: European Thyroid Imaging Reporting and Data System

Category	Score Tirads	Eu-TIRADS
0	–	–
1	1	EU-TIRADS1
2	2	EU-TIRADS2
3	3	EU-TIRADS3
4	4-6	EU-TIRADS4
5	7 & more	EU-TIRADS5

Several ultrasound features have been found to be associated to an increased risk of thyroid nodule cancer, the main ones being a cystic composition, a predominantly solid composition, hypo-echogenicity, size, shape (taller-than-wide), margin and the presence of micro-calcification (echogenic foci). Each features are assigned points ranging from 0-3 and the summation of these features' points determine its risk level. In order to standardize the ultrasound report that describes and evaluates thyroid lesions, an agreement which is called an European-Thyroid Imaging Reporting and Data System (EU-TIRADS) has recently established, see table (table 1 for more details. From table 1), one can study how nodules range according to the European-Thyroid Imaging Report And System-1 (benign) to European-Thyroid Imaging Report And System-5 (highly suspicious to be cancerous). A high score implies strong suspicion an the need for FNA (Tessler et al., 2018). Usually, thyroid nodules are heterogeneous, composed of various internal echo patterns that are confusing even

to experts. Eu-Tirads is a precondition for the Bethesda score system, a reporting system of thyroid cytopathology, which categorizes the nodules as benign, probably benign and malignant based on biopsy features, as shown in table 2).

The TIRADS score determines the risk level from US images, and helps making the decision on whether to perform a fine needle aspiration on the nodule or not. In clinical routine, if the TIRADS score is above the risk threshold, a fine needle aspiration process is taken on the thyroid nodule, and the Bethesda score, which is the most influential criteria in making the decision to perform surgery on the nodule or not, is calculated from the biopsy features. Stratification and estimating the Bethesda score manually is a tiresome and prone to variability task. Hence, we propose to build a Computer Aided nodule diagnosis system based on Bethesda scores to level the risk and avoid unnecessary surgical process on the patient.

Knowing the orientation of thyroid nodule ultrasound images is one of the important phases in 2D echography analysis. The orientation of a growing nodule is categorized as parallel (when the anteroposterior diameter of a nodule is equal to or less than its transverse or longitudinal diameter) or non-parallel (when the anteroposterior diameter of a nodule is longer than its axial or sagittal diameter). The orientation is categorized according to the relationship between the long axis of a nodule and the long axis of the thyroid gland, regardless of the nodule shape (Shin et al., 2016). For our local ultrasound images, we have two thyroid nodule orientation since two orthogonal views per case are acquired (sagittal and axial). Moreover, we made sure each image acquired contains only one nodule (Fig.1.

In this work, we proposed to develop a deep learning algorithm that uses thyroid nodule US images to decide whether a thyroid nodule should undergo a biopsy and to compare the performance of the algorithm with the performance of physicians who adhere to the European-Thyroid Imaging Reporting and Data System (TI-RADS). It is classically composed of four main stages, (I) Pre-processing, (II) Data augmentation, (III) Feature extraction and (IV) automatic classification of benign or malignant. The CAD system should ultimately eliminate the weaknesses of expert dependency, effort, time spend on investigation of nodule and lack of accuracy.

Our work has the following main contributions: 1) We proposed several pre-processing methods that help to enhance the image quality. The pre-processing steps that were implemented are noise removal, cropping, resizing, histogram equalization, and removing artifacts from the images. 2) We demonstrated that generating synthetic images can improve the detection result. 3) We integrated attention modules to the Densenet deep learning architecture, which brought substantial improvement to the classification results. The incorpo-

Table 2: The Bethesda System for reporting thyroid Cytopathology

Bethesda Category	Description	Risk of malignancy%	Managements
0	Benign	0	Normal
I	Undetermined	1 - 2	Repeat FNA
II	Benign	3 - 5	Follow-up
III	Follicular lesion	5 - 15	Follow-up
IV	Follicular neoplasm	15 - 30	follow-up
V	Suspicious malign	60 - 75	Surgical lobectomy
VI	Malignant	97 - 100	Total Thyroidectomy

Figure 1: Illustrative example showing the two orientations of thyroids nodules in ultrasound images. The white arrow points at flat shape that indicates in (A) a longitudinal view, while in (B) they point at round structures which are associated to a transversal view nodules respectively.

rated attention module specifically helps the network to focus on the strong features while estimating the malignancy. 4) We showed that computing focal loss with automatic assignment of loss weights based on the sample distribution of classes enables to overcome the data imbalance problem and improves the model performance with little rise for the computational cost. 5) We have compared several deep learning methods to overcome our dataset's main limitations which are its low image quality and its small size. 6) We illustrated interpretability of the classification of benign and malignant task using heat maps derived from Grad-CAM.

2. State of the art

A significant number of studies was carried out on this thematic area scientifically (Frates et al., 2005). Nodule detection studies can be classified into two main categories: non-machine learning based and machine learning based techniques. The non-machine learning are usually standard image processing approaches

with semi-automatic methods. It is mainly focused on thresholding the risk level by physicians. Most Machine Learning CADs are aimed to outperform experts' assessment accuracy. Basically, the studies involve the comparison of Computer Aided Diagnosis systems with the manual classification of the nodules by experts. We discuss hereafter the state-of-the-art for methods related to our work.

2.1. Classical machine learning algorithms for the detection of malignant nodules

In recent years, few machine learning methods have been proposed to diagnose the malignancy risk of nodules. In (Peng et al., 2017), the authors investigated the feasibility of applying the first order texture features to diagnose thyroid nodules in Computed tomography image (CT). A total of 284 thyroid CT images from 113 patients were used in this study. Their method involved the following steps: first, regions of interest (ROIs) were extracted manually by a physician. Second, some standard filters like median filtering were applied to reduce

photon noise before feature extraction. Third, a support vector machine (SVM) algorithm was applied to predict the classification task. The results of this paper work were measured using accuracy and sensitivity scores of 0.880, 0.821 respectively. Chi et al (Chi et al., 2017) presented a CAD system to identify as many malignant nodules as possible. The images used in this research work were from the following two datasets: **Database 1** is a publicly available thyroid ultrasound image database proposed by (Pedraza et al., 2015), consisting of 428 thyroid ultrasound images [1]. **Database 2** is a private database, consisting of 164 thyroid ultrasound images. A pre-trained GoogLeNet deep learning approach was used for feature extraction and a Cost-sensitive Random Forest as classifier to identify the malign nodule.

2.2. Deep learning algorithm for detection of malignant nodules

Deep learning design has showed a visible improvement in diagnosis of malign cancer from nodules. Most of the methods that have been implemented in this area use B-mode ultrasound images [2], as we do. (Buda et al., 2019) tackled the classification problem using their local dataset and a deep learning algorithm to provide management recommendations for thyroid nodules observed on ultrasound images, and compared its performance with physicians. They used 1278 nodules for training, and 99 nodules for testing. Their method used three main stages to accomplish the task using a Faster R-CNN network: First, they extracted the region of Interest (ROI) based on caliper markers localization. Secondly, they predicted the risk of malignancy using a multi-task CNN. Lastly, they built a stratification into risk level using the model. They showed that the performance of the algorithm was similar to that of the consensus of three expert readers. On the test set, deep learning achieved an Area under the curve (AUC) of 0.87 (95% Confidence Interval (CI): 0.76, 0.95), which is close to that of expert consensus (0.91; 95% Confidence Interval(CI): 0.82, 0.97). (Wu et al., 2016)'s study consists of 970 radiographical proven thyroid nodules from 970 patients. In this related work, a radial basis function (RBF)–neural network (NN) method was used as classifier. The deep learning method under-performed with respect to the experienced experts. Identification of malignancy by the experienced experts achieved the highest predictive accuracy of 88.66% with a specificity of 85.33%. whereas the radial basis function (RBF)–neural network (NN) achieved the accuracy of 84.74% with specificity of 76%. (Koh et al., 2020)'s research diagnoses

thyroid nodules from ultrasound images by ensemble of convolutional neural networks (CNNs). They collected datasets from multiple center, which amount to 15,375 US images of thyroid nodules. CNNs demonstrated higher area under the curves (AUCs) to diagnose malignant thyroid nodules (0.898–0.937 for the the internal test set and 0.821–0.885 for the external test sets) than the physician. AUC was significantly higher for CNNE2 than the one from physician decisions on their test set (0.932 vs.0.840). Recently, a few studies have been proposed to better classify nodules by involving an unsupervised learning method, called Generative adversarial deep learning network (Hang, 2021). This research work diagnoses thyroid nodules using images by the fusion of conventional features and residual-generative adversarial network (Res-GAN) features. Training sets come from an open-source thyroid nodule image dataset named "database of thyroid ultrasound images" (TDID). Most GANs nowadays are based on the Deep Convolutional Generative Adversarial Networks (DCGANs) architecture. The method which involves the combination of the deep features with the conventional features gives a promising performance in the model.

Focal Loss is a loss function that addresses class imbalance during training in tasks like image classification. It applies a modulating term to the cross entropy loss in order to focus learning on hard to classify examples (Lin et al., 2017). It is a dynamically scaled cross entropy loss, where the scaling factor decays to zero as confidence in the correct class increases. It has two hyper-parameters which are called alpha-α and gamma-γ. The focal loss introduces one new hyper-parameter, the focusing parameterγ, that controls the strength of the modulating term. When $\gamma = 0$, the loss is equivalent to the cross entropy(CE) loss. We define the focal loss in Eq. 1:

$$FL(p_t) = -(1 - p_t)^{\gamma} log(p_t) \tag{1}$$

Where FL is the focal loss, and hyper-parameter γ ranges from 0 to 5.

Attention Mechanism: It is well known that attention plays an important role in human perception, and so does it in artificial neural networks. The main purpose of the attention module is to automatically choose the most important intermediate features, and to carefully refine the best feature maps through the network. There are two well-known convolutional attention modules. The two sequential sub-modules are called channel and spatial attention. **Channel Attention** utilizes the inter-channel relationship of features maps. Every channel of a feature map is considered as a feature detector (Zeiler and Fergus, 2014). Channel attention multiplies the output after max-pooling or average-pooling with a shared network coefficients to scale feature maps. **Spatial Attention** creates a spatial attention map by ex-

---

[1] https://www.kaggle.com/datasets/dasmehdixtr/ddti-thyroid-ultrasound-images
[2] http://cimalab.intec.co/?lang=en&mod=project&id=31/

ploiting the inter-spatial relationship of features on the input images. The difference with channel attention is that spatial attention focuses on the locations where lot of information found, rather than on pondering whole feature maps.

3. Material and methods

3.1. Objective

Our objective was to develop an automatic thyroid ultrasound image classification system to prevent unnecessary fine needle aspiration (FNA). Benign-malignant nodule classification at early stage is a crucial step to prolong patient survival. The aim of this study is to propose a method for predicting nodule malignancy based on deep biopsy features. We came to achieve this general objective by tackling three main challenges. **First**, we have a small dataset to carry out the process of building a computer aided detection system. It is a challenge for classification due to the fact that a diagnosis task is very sensitive and usually needs plenty of datasets to train on. **Second**, the Image format is Joint Photographic Experts Group(JPEG). Since it applies lossy compression to images, this can result in a significant reduction of quality on the images. Also we do not have access to the image resolution as we would with Nifti or DICOM images. Hence, the dataset needs to be pre-processed in order to get important features from the images. **Third**, the target classes show uneven distribution of observation, as the negative (benign) class has more observation than the positive (malign) label.

3.2. Dataset

In this proposal, we used 595 US images of thyroid nodules in Joint Photographic Experts Group format coming from two sources. **Private Dataset:** It contains a set of thyroid Ultrasound images that includes a complete annotation and diagnostic description of thyroid lesions, using the Bethesda score (biopsy features) interpretation criteria. The images are labeled by experts from the Hospitals of Bastia and Dijon. Hence, the annotation criteria might be affected by inter-observer variation. The private database consists of **534** thyroid ultrasound images. 294 images from the aixplorer vendor have a size of 1440×1080, while 240 images have a size of 1280×960 as they come from the CANON vendor. 191 images in the database are labeled positive (with Bethesda score III to VI), while 343 images are labelled as negative (with Bethesda score = 0 or II).

Public Dataset is a publicly available and 61 thyroid ultrasound images has been used in our research from the public link mentioned on the footer. [3] .The images are in Joint Photographic Experts Group(jpg) format with different dimensions. All the cases are labeled

as malignant (positive) and it is an open access resource for the scientific community projects.

We split the validation set from the training set randomly. Out of these images, 473, 71 and 51 images are used as a training set, validation set and test set, respectively. The test set are chosen carefully from both vendors. For more details, see table 3 below.

3.3. Pre-processing

Images are collected from different ultrasound machines, leading to imbalance in exposure, size and other parameters. Before undertaking feature engineering, we should pre-process the raw images by including noise reduction and image enhancement in order to feed the model with better quality of images. Image enhancement is the procedure of improving the quality and information content of original digital images data before processing. We introduce several commonly used image enhancement techniques for our experiment, which are cropping, resizing, interpolation, histogram equalization, adding variability, normalization, removing artifact from images, and Gamma correction.

3.3.1. Normalization

Feature scaling is one of the most important data pre-processing step, the intensity of every patient image is normalized to have zero-mean and unit-variance. Algorithms that compute the distance between features are biased towards numerically larger values if the data is not scaled, so calibrating the details of images from different sources to the same scale is consequential.

3.3.2. Cropping and resizing

The images were firstly cropped and resized to have the same resolution which is the physical space represented by each pixel in the image, as shown in figure 2.
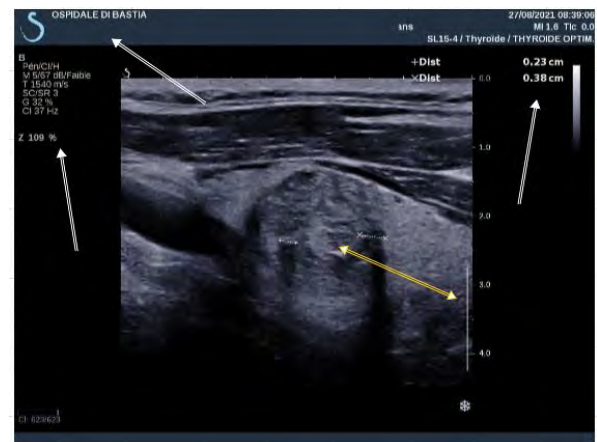

Figure 2: Ultrasound Image with noise and artifacts covering the textures.**White arrow:** indicates noises that appear on the images.**Yellow arrow:** indicates artifacts on the images

[3] https://www.kaggle.com/datasets/dasmehdixtr/ddti-thyroid-ultrasound-images

Table 3: The distribution samples in training, validating and testing groups of the dataset (I=Images), positive-samples =Bethesda -Score: 3, 4,5 and 6: negative-Samples =Bethesda-Score =0 and 2

Dataset	Samples (Bastia and Dijon)	Positive-Samples	Negative-Samples
Training	544 I	231 I	343 I
Val-Split	71	**Random**	**Random**
Testing	51 I	21 I	30 I
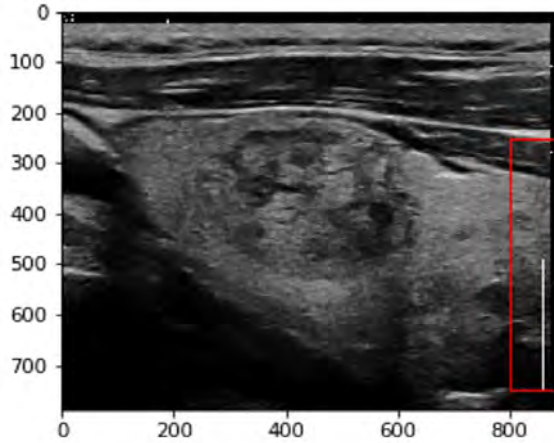
Figure 3: Pre-processed thyroid Ultrasound image and Details of how bounded by the red rectangle to remove the artifact

3.3.3. Histogram Equalization (HE)

HE usually increases the global contrast of many images, especially when the image is represented by a narrow range of intensity values. Through this adjustment, intensities can be better distributed on the histogram, utilizing the full range evenly. If most pixels are concentrated in the low gray area, the image will appear completely dark, but if they are concentrated in the high gray area, it will appear bright. Histogram equalization is therefore applied to elevate the contrast of the image, thus improving the visual effect of the image (Patel et al., 2013).

3.3.4. Removing Artifacts

In this step, we implemented an opening morphological operation to discard artifacts on the images. Artifacts are characteristics which appears in an image and which are not present in the original imaged object. They usually appear at the center or corners of medical images. A rectangle method was used to make bounding boxes around artifacts via an anchor point xy and its width, height and 4-connectivity method. We used a bounding box in order focus the kernel in some part of the images, otherwise we may loose essential information if this method is applied for the entire image. Following this, morphological opening was applied to remove small white thin lines from an image while pre-

serving the shape and size of larger objects in the image. We used a small structuring element to maintain the texture information, as shown in figure 4.

3.4. Data Augmentation

As mentioned above, one of the big challenge for this thesis work is to overcome the limitations of the dataset. We employed on-the-fly data augmentation, which allows transformed images to be produced from the original images with very little computation as the transformed images are not stored on disk. We used the TensorFlow ImageDataGenerator class to augment the images. Each generated image is randomly different from the original in certain aspects depending on the augmentation techniques. We do this by extracting random 800 $\times$ 600 patches from the various size of the images, and train our network on these extracted patches. At every iteration, batch size of transformed images are generated with different parameters like shifting, rotating, flipping, etc... Such image augmentation techniques not only expand the size of the dataset but also incorporate a level of variation in the dataset which allows the model to generalize better on unseen data,to observe the produced images, See (Fig. 5).
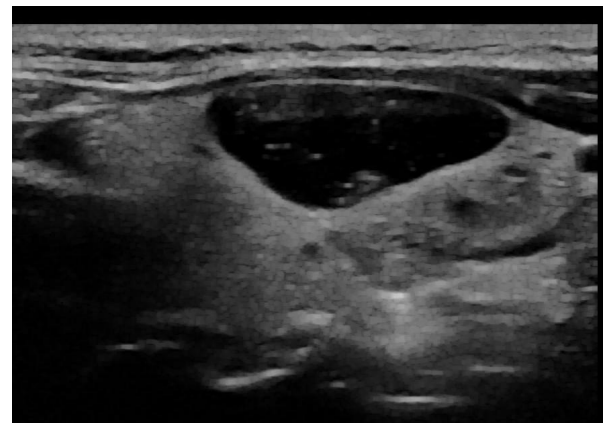

Figure 4: Ultrasound Image after pre-processing

3.5. System and Running

Recently, boosting the training to a satisfactory extent was achieved by using Graphics processing unit
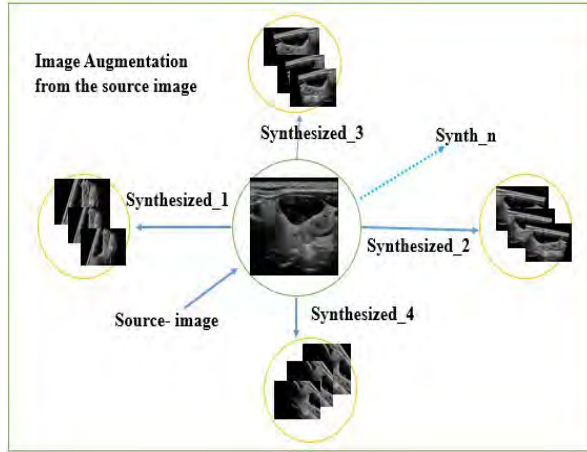
Figure 5: data augmentation: A source image is used to synthesize images like synthesized_1(rotation-range), synthesized_2(shift-range), synthesized_3(shear-range and shift-range) so on.US image respectively

(GPU)(Chen et al., 2014). This enhancement allowed us to efficiently utilize computational resources of the available GPUs and other software package tools in Imagerie et Vision Artificielle (ImViA). We have used persistence-m nvidia type of GPU with 12GB size of memory and CUDA version of 11.6 to launching the training for 320.40 min. We have used Ubuntu 20.04.4 LTS operating system and virtual environment with python version of python3.9 programming language.

3.6. Proposed Pipeline

The proposed pipeline consists of US image as inputs, pre-processing, data augmentation, deep learning based feature extraction, interpertability and thyroid nodule classification, as illustrated in (Fig.6). We compared several deep learning networks using this pipeline to different learning schemes. In our approach, the pipeline consists of an additional module. The module(attention and conv block) is incorporated with Densenet to extract features and classify malignant nodules, as can be seen from the figure see(Fig.6). The attenton module are integrated between convoutoinal layers. The conv block are appended at the attention module in the end for providing the output of the classification task.

3.6.1. Network Architecture

Convolutional neural networks have become the dominant machine learning approach for object classification (LeCun et al., 1989). We implemented three different deep learning techniques to tackle ultrasound thyroid image classification. First, we built a simple convolutional neural networks, and evaluated it on our dataset. Second, we implemented fine-tuned deep convolutional neural networks like Resnet-18, EfficientnetB0, and Densenet121. Lastly, we employed the

proposed deep learning architecture that incorporate an attention-conv module and Densenet. inside of it. The different architectures details are discussed as follow afterwards.

3.6.2. Convolutional Neural Network models

We first built a simple CNN model with twenty five (25) layers and let the neural network learn from scratch. This section briefly discusses the role of some components in CNN architectures. **Convolutional layers** are composed of a set of convolutional filters where each neuron acts as a feature detector and extracts feature pattern. **Pooling layer:** Once features are extracted, its exact location becomes less important as long as its approximate position relative to others is preserved. Pooling or down-sampling is an operation that sums up similar information and outputs the dominant response within this local region in order to compress information spatially. **LeakyRelu Activation Function** It is how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network.It adds non-linearity to the transformed outputs of layers. **Batch normalization** is used to address the issues related to the internal covariance shift within feature maps. The internal covariance shift is a change in the distribution of hidden units values which slows down the convergence(Ioffe and Szegedy, 2015). Data are scaled not only before entering training, but continues to stay scaled while it is training. **Dropout** introduces regularization to the network, which ultimately improves generalization by randomly skipping some units or connections with a certain probability. **Fully connected layers** are mostly used before the output for the final decision. It is a global operation and takes input from feature extraction stages to globally analyses the output of all the preceding layers. **Softmax Activation function** is used as the activation function in the output layer of neural network models in order to predict a binomial probability distribution.

3.6.3. Deep Convolutional Neural Network

DCNNs are a type of Neural Networks, which have deep layers and have shown exemplary performance on several competitions related to Computer Vision.We have used fine-tuned deep CNN architectures where layers are added to the trained model to adapt it for our task. We have tried different deep learning architectures that have been already trained on the ImageNet database(Deng et al., 2009). Hence, we got an opportunity to compare the architectures performance based on the result with our proposed methods. We have also confirmed that fine-tuned models work better than the model that we built and learn from scratch,See 5 for more details. The following pre-trained deep Convolutional neural network models have been used to perform classification in our task. **Residual Networks, or ResNet-18** is a convolutional neural network that
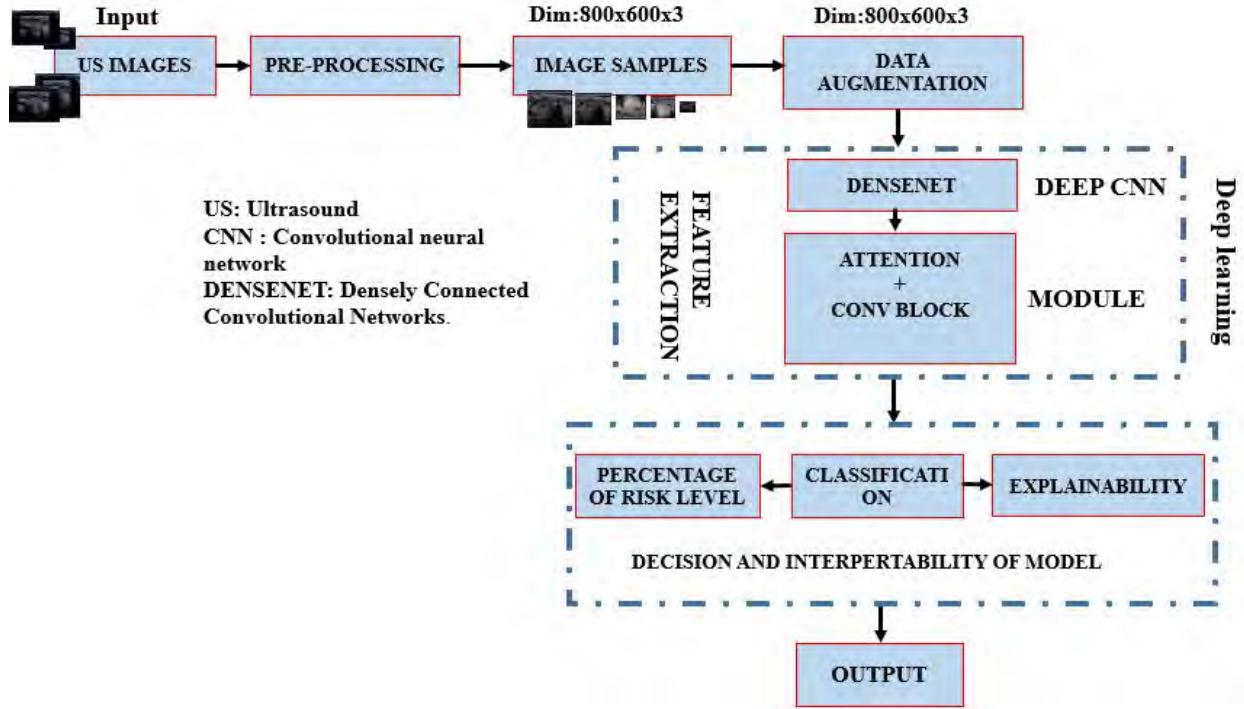
Figure 6: **Proposed pipeline**:Deep convolutional neural networks model for image classification, **800x600x3**: The dimension of image samples and its channel, **US**:Ultrasound image given as input for the model

is 18 layers deep. we have implemented the ResNet-18 architecture in two steps in the Tensoflow framework(Pang et al., 2020). we first discarded layers after the $18^t h$ in ResNet-50, then we added a block that we have designed to fit our problem and train the model. The block is composed two layers(ReLU activation function and GlobalMaxPooling2D) and one classifier function. ResNet-18 helps to overcame the vanishing gradient problem issue by introducing a so-called skip connections-that leaps over one or more layers(He et al., 2016). Some Layers were frozen to prevents the weights from being modified and random seed method is used to controlling random initialization of weights. **Efficient-netB0** is a deep convolutional neural network architecture with 237 layers. It uses a scaling method that uniformly scales all dimensions of depth/width/resolution using a compound coefficient. Unlike conventional practice that arbitrary scales these factors, the Efficient-NetB0 scaling method uniformly scales network width, depth,and resolution with a set of fixed scaling coefficients. EfficientNetB0 is a well known neural network architecture for compound model scaling methods though it does not work well for our task due to overfitting. In other words, scaling every dimensions balance all dimensions of the network depth, width, resolution and improves model performance (Tan and Le, 2019). We have set drop-connect-rate =0.4 during our demonstration.We happened two layers and one classifier function to this architecture to adapt the model to

our task.

Densenet121: It consists of 427 layers with 120 Convolutions and 4 average pooling layers. This network was designed to address the problem of vanishing gradient by directly connecting each layer to every other layer in a feed-forward fashion. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers and further exploits the effects of shortcut connections. Unlike residual neural networks (ResNets), the feature maps received from previous layers are concatenated not summed. Other than tackling the vanishing gradients problem(**?**). And also, a DenseNet network has translation layers between adjacent block and uses to update the size of feature-map through convolution and pooling layers. By knowing these all features of Densenet pre-trained model, We fine-tuned this model to our specific task and outperform other models that we mentioned them in this section. Hence, we propose an approach that integrated Densenet and module for the betterment of the result.

3.6.4. Proposed Architecture

We implemented several deep learning architecture to detect the malignant nodule.Densenets are efficient for classification tasks, because they have skip connections and better transmission of features across the network. Hence, we propose a method that incorporates Densesnet121 and module(attention+convolution block) for further improvement of the result. we chose

8

Densenet121 for further improvement for two reasons: First, Densenet architecture gave the the highest result comparing with the other models. Second, this network does not suffer with the problem of vanishing gradient. One simple interpretation of this is that the output of the identity mapping was added to the next block, which might impede information flow if the feature maps of two layers have very different distributions(Simonyan and Zisserman, 2014). We proposed an approach that incorporates a module(attention+conv blok) to Densenet121's architecture, intending to enhance the performance of the networks. There are two types of attention module 2, and we have arranged them in a sequential manner(Woo et al., 2018). We have used channel attention first and then spatial attention,as illustrated in (Fig.7), because of this arrangement gave the better result, than the spatial-first chain. The attention modules are integrated between convolutional layers to refine the most important feature maps, as illustrated in figure(Fig.9). Channel attention helped the proposed model to concentrate the substantial information of the input US image. Spatial attention brings focus to specific parts of spatial information, pondering feature maps to enhance regions of interest on the images. The attention module down-sampled feature maps using average and max-pooling operations.Therefore, the attention mechanism played a great role in guiding the networks to concentrate on the most important feature maps.

We have also add a block that consists the succession of layers: Separable convolution, batch Normalization, GlobalMaxpooling2D, rectified linear activation unit, dropout, a fully connected layer and finally a Softmax classifier function. See the whole network in (Fig.**??**). **separable convolution** is a kernel in which a single convolution can be divided into two or more convolutions to produce the same output with a much lesser computation cost. **GlobalMaxpooling2D** used to reduce the dimensionality of the feature maps and give one maximum value for a whole region to strongly compress information.The rest layers have been explained in this(Sec 3.6.2) section. Basically, We appended the block to the proposed network to minimize the covariance shift problem and let the network learn representation pattern our dataset. Because, the block contains batch normalization, separable conv,etc Therefore, the proposed method includes: Densenet121, attention module and convolutiona blocks and a classifier function.

4. Optimizer

We used the adaptive moment estimation (ADAM) optimizer to monitor the training and optimize the convergence when training the model.

4.0.1. Loss Function

We used loss function to optimize the parameter values in the proposed neural networks. Basically, It is a method of evaluating how well our model get well with the the given input data. As an objective function, we have used two different loss functions to evaluate our model by computing the error. **Categorical Cross Entropy Loss** is the probability value among the given classes for a classification task. Cross-Entropy calculates the average difference between the predicted and actual probabilities, as explained in this (Eq. 2) equation.

$$\mathbf{L} = -\sum_{i=1}^{N} y_i log \hat{y}_i \qquad (2)$$

where $\mathbf{L}$ is loss, N is output-size the output size, $\hat{y}_i$ is the i-th scalar value of the model output, y_i is the corresponding target value, and the output size is the number of scalar values in the model output. Categorical Cross Entropy Loss is not recommended to be used as error optimizer for imbalanced dataset. Hence, we proposed another loss function which is called focal loss, for more detailed, (Sec 2) section. Focal loss is the modification of cross entropy loss and have two hyper-parameters. while α balances the importance of positive/negative examples, γ tries to penalize the misclassified examples. As γincreases, the shape of the loss changes so that easy examples with low loss get further discounted. We tried γ values from 1 to 5 and observed that model's classification accuracy increases with γ values. However, It became almost constant after γ reached 3.5 in our case. According to our experiment, Focal loss works well and helps diminishing the impact of data imbalance compared to Categorical Cross Entropy Loss, see (table 4) for more detail.

4.1. Performance Evaluation Metrics

After applying the deep learning algorithms, evaluating the model is very important to know how the system behaves on unseen data. A tool which is called a metric is introduced to measure the accuracy of the models. In this paper, we use several common metrics for classification problems to obtain valuable information about the performance of algorithms and to run a comparative analysis. These metrics are accuracy, f1-score, confusion matrix and classification report. We rarely used sensitivity and specificity in the evaluation mechanism as confusion matrices are easier to interpret.

4.1.1. Accuracy

It is the most used and maybe the first choice for evaluating an algorithm performance in classification problems. It can be defined as the ratio of accurately classified data items to the total number of observations, see (Eq. 3). Despite the widespread usability,
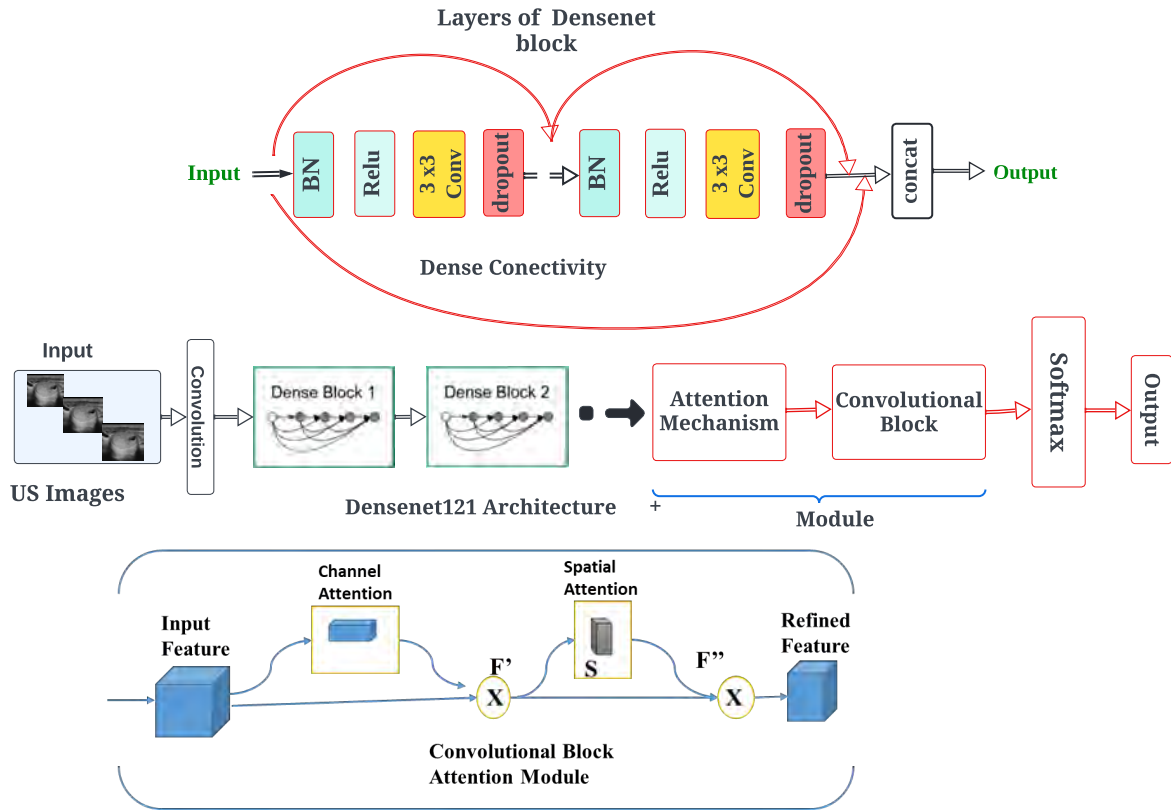
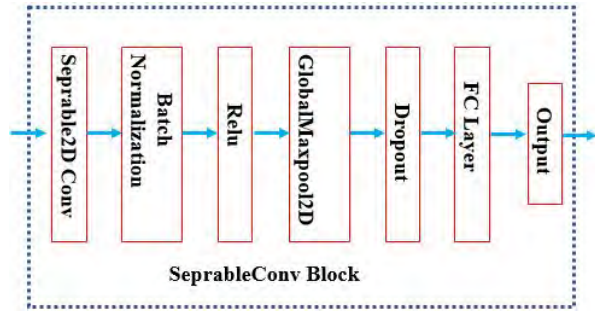Figure 7: Proposed DenseNet Architecture with concatenated attention module and block

Figure 8: Convolutional block that consists of six top layers

accuracy is not the most appropriate performance metric in some situations, especially in the cases where target variable classes in the dataset are unbalanced (Vakili et al., 2020).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \qquad (3)$$

TP, TN, FN and FP represent the True Positive, True Negative, False Negative and False Positive of predicted image respectively. Basically,It is the summation of TP and TN which are correctly classified over the total datasets.

4.1.2. F1-score

This metric, which is also known as f-score or f-measure, takes both precision and recall into consideration in order to calculate the performance of an algorithm(Goutte and Gaussier, 2005). Mathematically, it is the harmonic mean of precision, see (Eq. 4a) and recall, see (Eq. 4b) formulated as follows (Eq. 5):

$$precision = \frac{TP}{TP + FP} \qquad (4a)$$

$$Recall = \frac{TP}{TP + FN} \qquad (4b)$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \qquad (5)$$

We observed that accuracy metric does not work well with data imbalance condition, Since it does not distinguish between the numbers of correctly classified images of different classes. F1-score is a proper measure when working on classification tasks in which the data points are imbalanced.

4.1.3. Confusion Matrix

This matrix is one of the most intuitive and descriptive metrics used to find the accuracy and correctness of
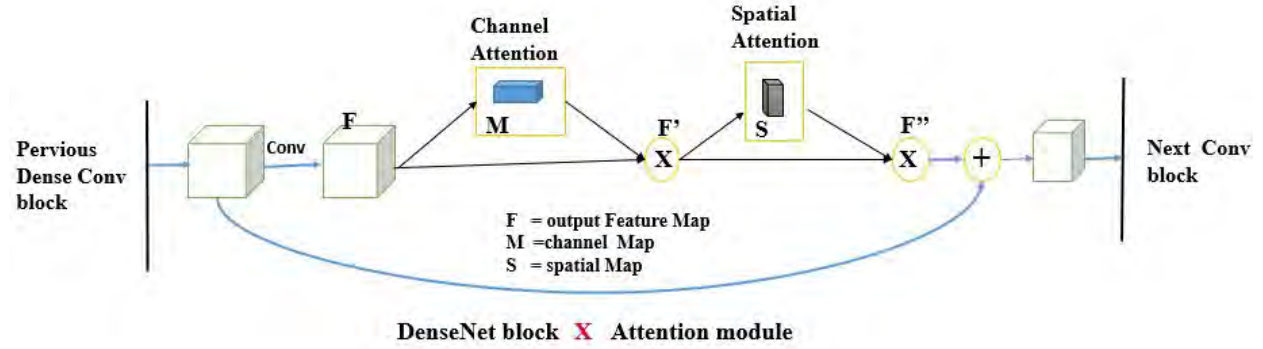
18.10

Figure 9: Attention module integrated with in a DenseBlock in DeseNet Densenet architecture, **F** indicates the refined outputs feature map

a machine learning algorithm. Its main usage is in classification problems where the output can contain two or more types of classes (Townsend, 1971). We can compute sensitivity focusing on the True positive rate and specificity focusing on the false positive rate from the confusion matrix.

4.1.4. Classification Report

Classification report is an evaluation metric in deep learning machine learning. It is used to display precision, recall, F1-score and support for the trained model as package. Support is the exact number of occurrences of each class in the specified testing dataset.

4.2. Thyroid Nodule Risk Level Assessment

The estimation of the risk level from ultrasound imaging of thyroid nodules is extremely difficult. In this thesis work, we provided a probability expressed in percentage(0% to 100%) to represent the malignancy level. This would be very helpful for physician to take decision in the need for fine needle aspiration. For instance, there are some images that obtain a score between 40% and 60%, which we call "gray zone" associated to an uncertain prediction, and which would need strict follow up of treatment. If the image get a score above 61% of benignity, it is in normal status. Otherwise, it might be needed to take a serious measurement or follow up at the patient.

4.3. Interpretability

Deep neural networks have been widely-known for their magnificent performance in playing with different machine learning tasks. However, because of their exceeding-parameterized "black-box" nature, it is usually back-breaking to understand the prediction results of deep models(Dong et al., 2017). Interpretability(Explainability) is the degree to which a human can realize the cause of a decision and outputs can be described in the way that make sense to deal with deep understanding how a model makes prediction. It also helped us to debug the network. In our proposed

method, we have used the gradient-weighted class activation map (Grad-CAM) to give insights on the decision making (Selvaraju et al., 2017). Grad-CAM uses gradients to give a coarse localization map highlighting the most substantial regions in the input image when predicting the result. The class activation map simply indicates the discriminative region in the image which the CNN uses to classify that image in a particular category. We can identify the importance of the image regions by projecting back the weights of the output layer on to the convolutional feature maps. A graphical representation which is called heatmap method is responsible for highlighting the discriminative region used by the model.

4.4. Training

Weights were initialized using *He normal* initialization method (He et al., 2015). It draws samples from a truncated normal distribution centered on 0 The optimization of the weights are done using Adam as the optimizer with learning rate of 0.0001. The minibatch size was 8, because a small batch size is recommended for small datasets. The models were trained until convergence for various numbers of epochs including depending on the model. We empirically selected a weighting factor of 0.50 for α, and 3.50 for γ, the hyperparameter in of the focal loss. For above 200 epochs, we were using α =0.50 and γ =2.0, as the fact γ hyperparameter has reciprocal relationship with number of epochs. We used the programming language Python and the library Tensorflow to implement the deep learning models. We fixed the random seed to 42 to set the integer starting value used in generating random numbers. Setting random seed to fixed value is very important so as to get stable or gives reproducible result with Tensor-Flow framework.

In order to avoid over-fitting, we adopted three techniques: dropout, early stopping and data augmentation. Dropout is a regularization technique where randomly selected neurons are dropped during training. The ignored neurons will not have contribution during a forward and backward propagation. Dropout reduces over-

fitting by preventing complex co-evolution on the training data. In our all experiments, we used dropout with a probability of 0.25.

During training, the training and validation losses decrease, usually in the staring the training and validation loss decreases. As the number of epochs increases, the training loss will continue to decrease but the validation loss will slowly diverge over time. This phenomenon indicates overfitting, which does not generalize well to unseen data. To monitor this, we used early stopping techniques, which triggers when the validation loss starts to increase. The training immediately stops after certain number of epochs, to give the possibility to the validation loss to decrease again, in case the training curves are noisy. In our experiments, the patience parameter for the early terminating of the training process was 20 epochs.

The last techniques that we employed to handle overfitting is data augmentation. A lot of similar images were synthesized synthesized applying transformations such as shearing, rotating,zca-whitening,etc. This helps to artificially increase the dataset size, which helps avoiding over-fitting. The reason for it is that, as we generate more data,the model can not learn by heart the training data and is forced to learn generalizable features and give good performance on unseen data.

5. Experiment and results

We conducted experiments with the models, and training schemes previously mentioned in the methodology section,(Sec (3). The proposed network outperform other networks. To evaluate the classification results of thyroid nodule, we used Accuracy, F1-score, confusion matrix, specificity and sensitively metrics. To evaluate our models, we split the dataset between train, validation and test sets, as done traditionally. The validation set is drawn from the training data, it is kept aside the optimization to set hyper-parameters and to detect overfitting. Train-validation-test evaluation methods as well as a validation set that are separate section from the training dataset to get evidence how well the model is performing on images that are not being used in training.

To evaluate the effect of pre-processing in our method, we compared the results with and without the mentioned pre-processing steps. when we say Unprocessed image, images are given as raw data and not scaled for the proposed method. This direct classification of malignancy from the full-sized unpre-possessed thyroid ultrasound images(Fig.2). our method yielded accuracy and F1-Scores of 0.772 and 0.813 respectively. The model suffer from overfitting problem due to the noisy and unrepresentative training data. In the sense that model is learning a detail of noise in the training data to the extent of it negatively impacts the performance of the model on aw data. However, when we

employed pre-processing, our method achieved an improved accuracy and F1-scores of 0.9007 and 0.9216 respectively.

We can observe the effect of changing the loss function by looking at the diagram in (Fig13), and (table 4). We can see that the Focal Loss helps to deal with a limited and imbalanced dataset. Especially, the loss curves are less noisy with focal loss than with cross entropy loss function. Focal loss is used often with hyper-paramaters of $\alpha = 0.50$ and $\gamma = 2.0$. but we used other values after tuning which ranges from $\gamma = 1$ to $\gamma = 5$.

Table 4: Quantitative comparison of loss functions using accuracy and F1-score with proposed approach, ±:shows small variation of the value per training

Loss Function	Accuracy	F1-Score
Cross-entropy	0.850 ± 0.030	0.840 ± 0.045
Focal Loss	$\mathbf{0.8700 \pm 0.0250}$	$\mathbf{0.9005 \pm 0.0216}$

We also demonstrated that adding synthesized images improves the performance of the model effectively. The results of the proposed model with and without data augmentation, and with a batch size of 8, are shown in this table (Table5).

Table 5: Classification results of the proposed architecture, ±: shows small variation of the value per each training

Metrics	Proposed w/o augmentation	Proposed with augmentation
Accuracy	0.701 ± 0.041	0.870 ± 0.021
F1-Score	0.750 ± 0.037	0.900 ± 0.021
Specificity	0.694 ± 0.036	0.850 ± 0.022
Sensitivity	0.802 ± 0.028	0.9300 ± 0.0342

We compared our proposed approach with four different networks regarding training time and performance. We can see that the proposed architecture works better than the other architectures on a number of the same experiments, see(table 6). Hence, Densenet incorporated with our attention module outperforms other neural networks when considering all metrics.

For the 51 test nodules, the proposed deep learning algorithm outperform the reported results of the previous related research works, as can be seen in this table (Table 7). Taking into account the inter-observer variation of manual identification of malignancy, the overall accuracy of the proposed deep learning method for thyroid nodule biopsy recommendations is better than the experts, as shown in (Sec 2). These results, however
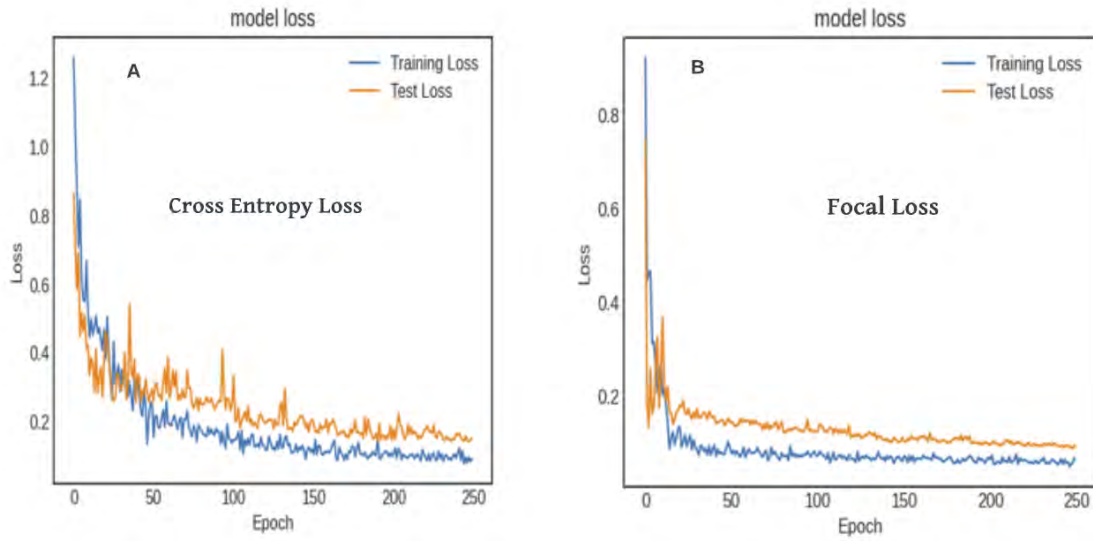
12

Figure 10: Qualitative comparison of loss functions on monitoring the proposed model during training. A) Cross-Entropy loss, B) Focal loss

promising, were obtained on a small dataset, and can not be compared directly to other studies.

We compared the proposed method with fine-tuned Densenet121 network, and three other networks which employed in the same pipeline except the modules. The module(attention-conv block) has only integrated with the proposed approach. The hyper-parameters were set to fixed value during the comparison. But, we got different results from each approach due to variation in size, behaviour and structure of the network architecture. We observed that significant changed can be achieved in the result by modifying the networks. The new method is proposed on a little modification of Densenet architecture, which is consolidating a module within it. The proposed method outperform others in all experiments. This is due to the fact that attention method and con-block are playing good role in extracting the most important features. The comparison has been done both quantitatively and qualitatively,as shown in (table 6 and (Fig. 11 respectively. EfficientNet did not perform well on both classes comparing with the other models due to the degradation problem. We obseved that EfficientNet model tried to ends up memorizing the data patterns and put up with random fluctuations. We can say that this model is suffering with gradient vanishing problem when it trained with our dataset. Hence,it has low performance on the test set with average F1-score of 0.7418. The model which is built and learnt from scratch has performed well, but a bit less than ResNet-18. Because, the model sufferers with model complexity due to the huge number of parameters. ResNe-18 has good performance on the on the classification task due to its size and special structure for handling gradi-

ent vanishing issue, but it is biased to positive samples. Well, its overall result is not promising like Densenet, and the proposed method. Densenet is very efficient in handling and reusing features maps with dense connections. And also, it has a translation layers that helps to update the size of feature-maps through layers. It performs well next to the the proposed method. We added very important module to the Densent that consists attention and conv block. The main task of this incorporated module, is to guide the architecture to focus on the most substantial features. The output of feature maps from dense convolutional layer is given as input for these module to downscale and forward as output for the next convolutinal layers, as illustrated in (Fig 9). The proposed method showed high performance with accuracy of .9007 and F1-score of 0.9216. We made qualitative comparison between the fine-tuned Densenet and proposed technique, illustrated here(Fig 13).

As shown in (table 6), we also compared the number of trainable parameters of the architectures. ResNet-18 has the lowest number of trainable parameters which is 0.25 million. The proposed method have has 0.75 million parameters. While the CNN has the highest number of parameters with 15.4 millions, which can be explained with the high numbers of parameters for fully connected layers. On top of that, the proposed method can be used to estimate the malignancy risk as illustrated on (Fig 16) by exploiting the non-binarized output. As we can see from the (Fig: 17), the confidence interval is 91% percentage. This indicates that the nodule is in normal condition (not cancerous). This estimation gives an important information for the physician to interpret the automatic prediction and to take the required treat-

13

Table 6: Accuracy and F1-Score comparison of various methods for thyroid nodule classification

Methods	Accuracy	F1-Score	No of Parameters
CNN	0.750 ± 0.045	0.770 ± 0.034	15.4 million
EfficientNetB0	0.760 ± 0.014	0.740 ± 0.018	4.7 million
ResNet18	0.840 ± 0.020	0.800 ± 0.034	0.25 million
Densenet121	0.880 ± 0.024	0.860 ± 0.019	7.1 million
Proposed Method	**0.870 ± 0.037**	**0.900 ± 0.0216**	**0.75 million**
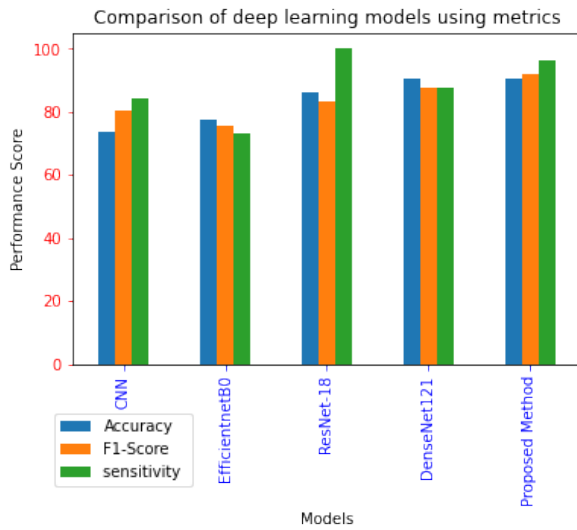
Figure 11: Comparison of models using barplots with maximum value scored by the models

ment on the patient. This is done based on the probability distribution of being cancerous from 0% to 100% using outputted by the softmax activation classifier, as this score can be interpreted as the model's certainty in its prediction.

The model is classifies the images by looking at some parts of the image. For some images, it is able to look at the centre part of the images, except in a few benign images. In this thesis work, We illustrated a Visual explanations of deep Networks using gradient weighted class activation maps. We observed that the model made classification by extracting the information from center region of the image as might be expected. See(Fig.18) By the using gradient-weighted class activation maps(Grad-Cam), we are able to interpret the reason behind the misclassified images, which is really wonderful and gives a substantial hints for further amendment of the network. The localization of instance has done from the final convolutional layer. The model is not looking at the right part of the the images, See(Fig.19). The reason may be because the nodules are a lot bigger in these images than on the usual ones. This

may further be improved by adding more cases such as these.

6. Discussion

In this thesis work, we evaluated our proposed pipeline and proposed method network for classification task on compressed ultrasound image which has two orientations per case. The dataset is very small (595 images) and has poor quality (contains a lot of noise and artifacts), which however corresponds to what experts use.

We found in our experiments that the distribution of image cases (malignant vs benign) in the training group was imbalanced. Therefore, we fused two databases (Private and public) to have enough training samples for both classes. To overcome these limitations, we proposed a pipeline that consists of pre-processing, augmentation, feature extraction, and classification task. Image pre-processing proved to be effective in improving the proposed method: (1) Cropping and resizing the acquired images in order to remove the different noise within the images. (2) Discarding of the artifacts uses to keep away the network from learning meaningless information and recuperate the textures overlapped by the markers made by the experts or physician. This is done using morphological operation using 3 x 3 kernel. (3) Histogram Equalization is contrast adjustments techniques that effectively spreading out the most frequent intensity values throughout the image. (4) the image normalization that adjust the details of images from different sources imaging techniques to the same scale. And also, we have added a contrast variability to the images. Then, the images are well cleaned and have good quality for further process.

As mentioned in Section (3), one of the main problem was the size of the dataset. In spite of getting training examples with higher quality of extracted features, the millions of parameters available 6 to be tuned for the network still need a huge number of examples to prevent the over-fitting. We came to solve it with image data generation techniques. The input image size was 800 x and 600, and we augmented the image to have enough
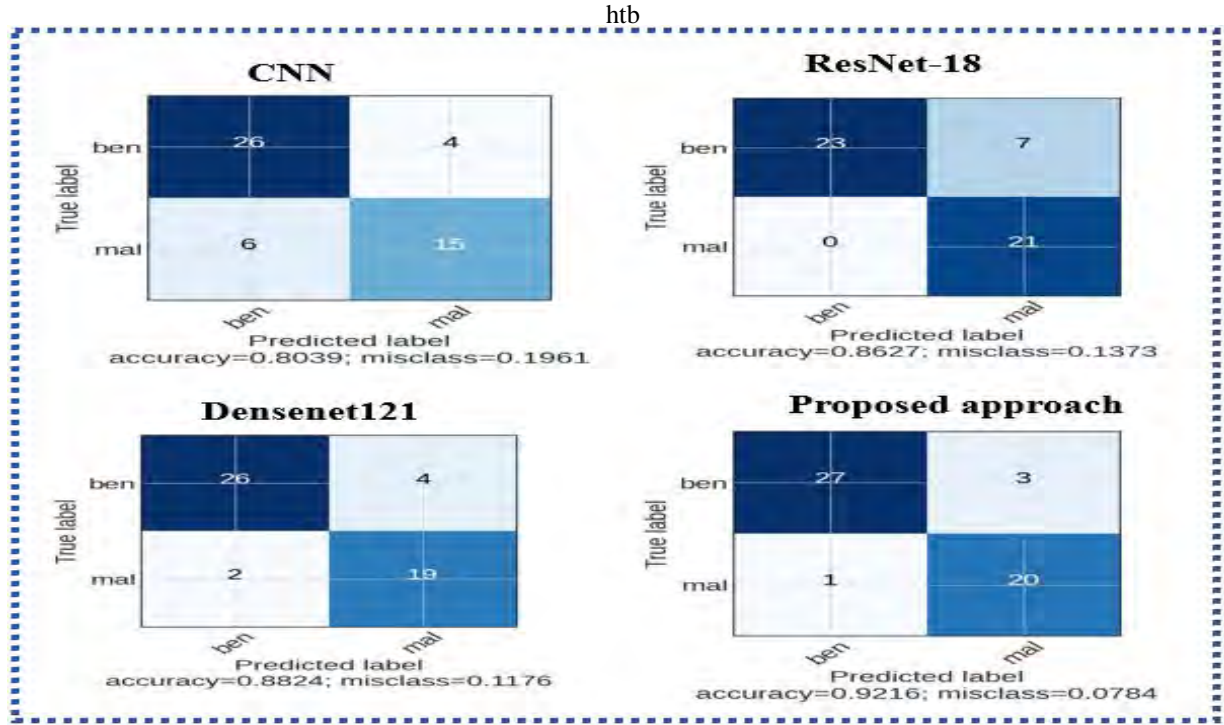
14

htb

Figure 12: Qualitative comparison of different models on classification of thyroid nodules. **A**: Confusion matrix of CNN with 10 misclassified examples, **B**:Confusion matrix of Resnet-18 with 7 misclassifications, **C**: Confusion matrix of Densenet121 with 6 misclassifications, **D**: Confusion matrix of the proposed method with 4 misclassified examples

number of training samples. The performance of the model might be affected due to cropping technique of the images.

From our experiments, the proposed method proves to have the advantage of needing less training samples to generate a CAD system based on deep learning networks. The system can help in minimizing the time, effort of physician and avoids unnecessary fine needle aspiration on patients. Regarding the use of loss function, the pipeline achieved better result with focal loss in terms of Accuracy and F1-Score comparing with categorical cross entropy loss. This indicates that how focal loss can improve the result by reducing the false positives, false negative and mitigating the class imbalance problem between training samples. Focal loss is efficient to classify the hard examples using penalized learning method. This the most recommend loss function for a research work that involves with data imbalance problem. Automatic assignment of weight to each class play good role in handling of class imbalance as well. We have tried two way of class weight assignment methods. 1) manually assignment high weights for the minority class. 2) automatic assignment of weights based on the distribution of dataset. According our demonstration, the second method works well and can be an hypothesis to tackle unbalanced data. Our research work was restricted in only classifying of thyroid nodules from Ultrasound images (US) into two classes of probably malignant and benign. We do not yet

have enough examples with all the 7 different Bethesda scores(0-VI) to attempt a model that can make prediction per each class. The dataset we have used are labeled by experts. Having this in mind, our classification task is still highly dependant on the experience of the experts and their subjectivity in interpreting ultrasound images of thyroid nodules. Our method could have a significant benefit in helping experts during the annotation process.

The performance of the proposed method based on deep learning networks is much improved in compare with state of the art and other deep learning architectures in classification of thyroid nodule task. In this (table 6), we did a comparison of our approach with other deep learning schemes. EfficinetNetB0 had the worst classification performance. The main reason is that it became overfitted the on the small dataset very fast. The proposed method that uses the added module to focus on the most relevant feature maps achieved better its results. We also compared the number of trainable parameters. Resnet-18 has the lowest number of parameters due to the reason that is has fewest layers than others. Densnet encourages feature reuse which substantially reduces the number of parameters, but still has high parameters due to its huge structure size. From this research work, We can suggest that our computer aided diagnosis tool can be used in Integrated Healthcare system(IHS) to help physician to early and accurate classification of nodule. we had some challenges and can be hypothesized in some way for further investiga-
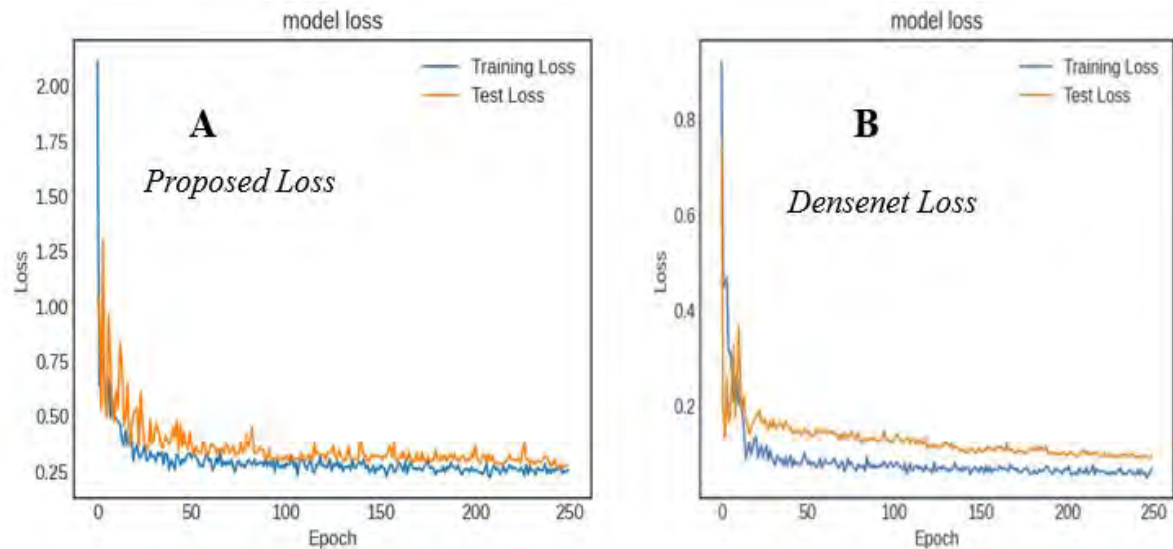
15

18.15

Figure 13: Qualitative comparison of loss functions fine-tuned Densenet and Proposed method monitoring the model during training:A) Loss function of Proposed model, B) Loss function of Densenet model

Table 7: Comparison of different CAD system in classifying benign and malignant thyroid nodule images

Methods	Learning Algorithm	Accuracy	Sensitivity
(Buda et al., 2019)	Faster R-CNN	83.00%	87.00%
(Wu et al., 2016)	(RBF)–neural network	84.74%	92.31%
(Peng et al., 2017)	SVM(Kernel=RBF)	88.00%	82.10%
(Koh et al., 2020)	InceptionResNetv2	85.00%	91.80%
Proposed model	**Attention-Densenet121**	**90.70%**	**96.42%**

tion. We do not know where the nodules are exactly on the labeled images. So, the annotation of ground truth can be done directly on the images with help of experts. This could give a comfortable environments to extract the region of interest(ROI) from the images. This could improve the performance of the our CAD syetem. And also, semi-supervised techniques can be used to annotate unlabeled US images In spite of data acquisition for this classification task is in ingrowing, auxiliary classifier Generative Adversarial Network(acGAN) can be used to generate synthetic training samples as it required. Future improvement that involves providing of the thyroid nodules which are transversal and longitudinal to the deep learning algorithm as it could provide additional gains in performance and classification results. This task could be incorporated with Generative adversarial network(GAN) to have enough two view training samples.

7. Conclusion

In this work, we proposed a deep learning based Computer aided diagnosis system for automatic classification of thyroid nodule disease from ultrasound images.We demonstrated several deep learning approaches and able to compare them in our method in same dataset. Our method uses incorporated module within the DenseNet architecture,and we showed that adding this module to the fine-tuned Densenet121 substantially improves the classification result. We have shown that pre-processing and augmenting effectively improved the the performance of our proposed model. Despite having a small size, heterogeneity, unbalanced, low image quality, our approach obtained overall good result on the test set achieving an accuracy 0.9007 and a F1-score of 0.9216 for detection of nodules, which is higher than the performance reached on recent studies on this thematic area. This method could be used to predict nodule malignancy in clinical practice two reasons where it could bring the following benefits. First, it can eradicate the

Figure 14: Prediction of malignancy with 96% confidence

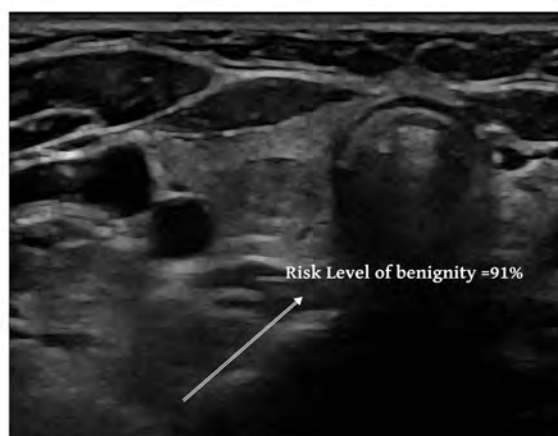Figure 16: Prediction of malignancy with 77% confidence

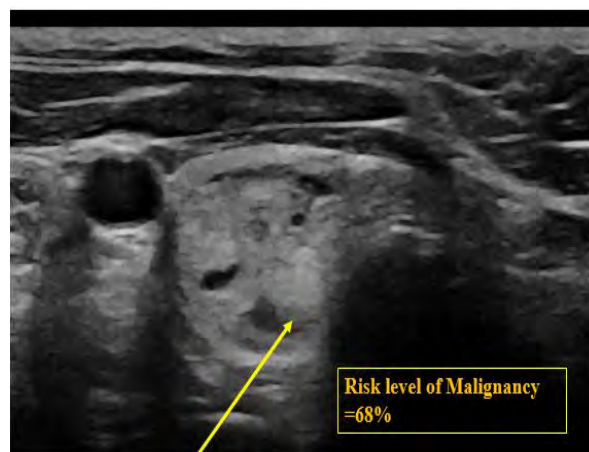Figure 15: Prediction of benignity with 91% confidence

Figure 17: Prediction of benignity with 68% confidence

substantial inter-reader variability, and subjectivity that have been noticed for this task even when a standard Interpretation criteria is used. Second,the proposed approach could reduce the time and effort that is required for analyzing thyroid nodules, which would be of great help for clinical experts.

Furthermore, We have used Gradient-weighted class activation maps(Grad-cam) method to provide an explainable heat map of the primary regions of interest used by the model of the proposed technique. It helps to visualize how the model make decision during the prediction.

8. Acknowledgments

References

Buda, M., Wildman-Tobriner, B., Hoang, J.K., Thayer, D., Tessler, F.N., Middleton, W.D., Mazurowski, M.A., 2019. Management of thyroid nodules seen on us images: deep learning may match performance of radiologists. Radiology 292, 695–701.

Chen, Z., Wang, J., He, H., Huang, X., 2014. A fast deep learning system using gpu, in: 2014 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE. pp. 1552–1555.

Chi, J., Walia, E., Babyn, P., Wang, J., Groot, G., Eramian, M., 2017. Thyroid nodule classification in ultrasound images by fine-tuning deep convolutional neural network. Journal of digital imaging 30, 477–486.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee. pp. 248–255.

Dong, Y., Su, H., Zhu, J., Zhang, B., 2017. Improving interpretability of deep neural networks with semantic information, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4306–4314.
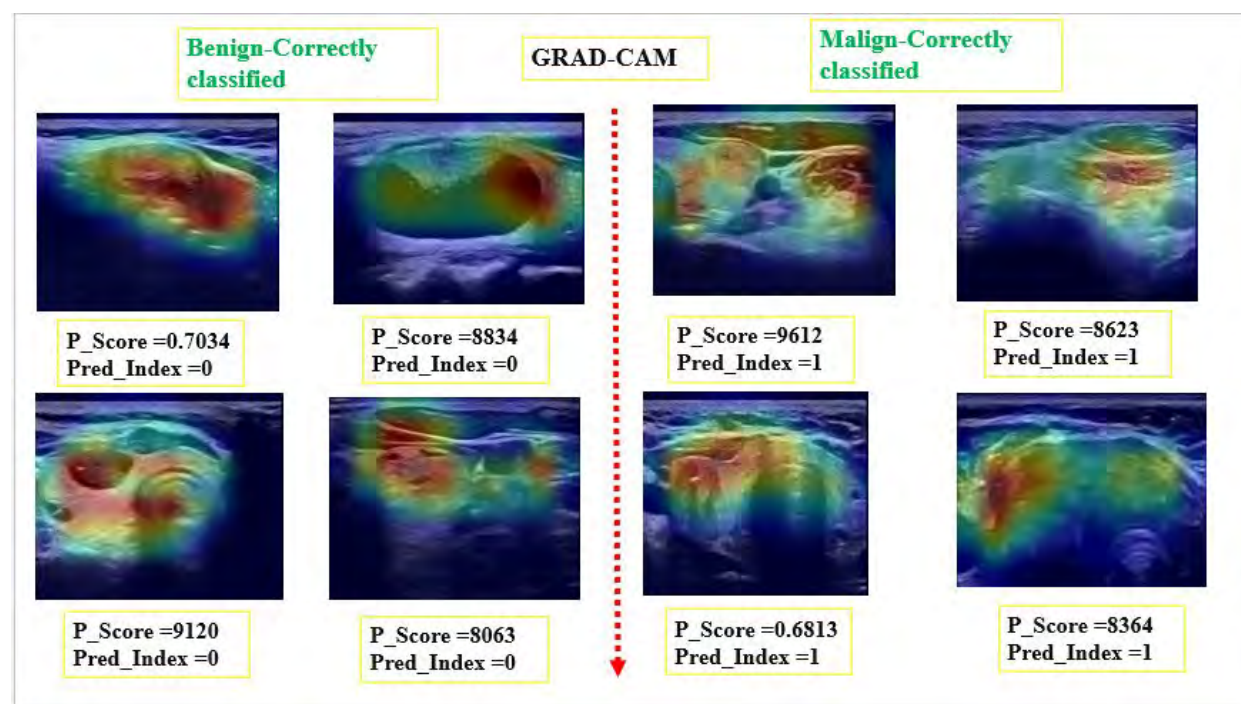
Figure 18: Illustration of correctly predicted images using Grad-CAM:**P-score:** probability score,**Pred-index:** prediction class
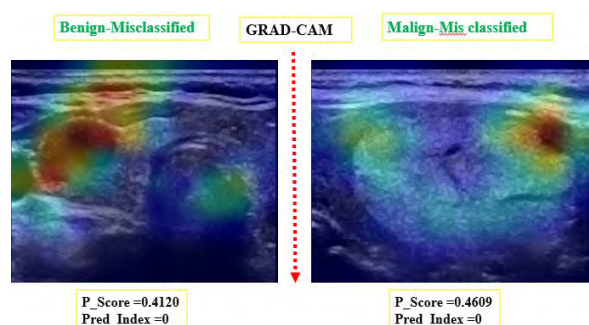

Figure 19: visualization of wrongly predicted images using Grad-CAM:**P-score:** probability score,**Pred-index:** prediction class

Frates, M.C., Benson, C.B., Charboneau, J.W., Cibas, E.S., Clark, O.H., Coleman, B.G., Cronan, J.J., Doubilet, P.M., Evans, D.B., Goellner, J.R., et al., 2005. Management of thyroid nodules detected at us: Society of radiologists in ultrasound consensus conference statement. Radiology 237, 794–800.

Goutte, C., Gaussier, E., 2005. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation, in: European conference on information retrieval, Springer. pp. 345–359.

Hambly, N.M., Gonen, M., Gerst, S.R., Li, D., Jia, X., Mironov, S., Sarasohn, D., Fleming, S.E., Hann, L.E., 2011. Implementation of evidence-based guidelines for thyroid nodule biopsy: a model for establishment of practice standards. American Journal of Roentgenology 196, 655–660.

Hang, Y., 2021. Thyroid nodule classification in ultrasound images by fusion of conventional features and res-gan deep features. Journal of Healthcare Engineering 2021.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, pp. 1026–1034.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International conference on machine learning, PMLR. pp. 448–456.

Koh, J., Lee, E., Han, K., Kim, E.K., Son, E.J., Sohn, Y.M., Seo, M., Kwon, M.r., Yoon, J.H., Lee, J.H., et al., 2020. Diagnosis of thyroid nodules on ultrasonography by a deep convolutional neural network. Scientific reports 10, 1–9.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., 1989. Backpropagation applied to handwritten zip code recognition. Neural computation 1, 541–551.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Pang, B., Nijkamp, E., Wu, Y.N., 2020. Deep learning with tensorflow: A review. Journal of Educational and Behavioral Statistics 45, 227–248.

Patel, O., Maravi, Y.P., Sharma, S., 2013. A comparative study of histogram equalization based image enhancement techniques for brightness preservation and contrast enhancement. arXiv preprint arXiv:1311.4033 .

Pedraza, L., Vargas, C., Narváez, F., Durán, O., Muñoz, E., Romero, E., 2015. An open access thyroid ultrasound image database, in: 10th International Symposium on Medical Information Processing and Analysis, International Society for Optics and Photonics. p. 92870W.

Peng, W., Liu, C., Xia, S., Shao, D., Chen, Y., Liu, R., Zhang, Z., 2017. Thyroid nodule recognition in computed tomography using first order statistics. Biomedical engineering online 16, 1–14.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, pp. 618–626.

Shin, J.H., Baek, J.H., Chung, J., Ha, E.J., Kim, J.h., Lee, Y.H., Lim, H.K., Moon, W.J., Na, D.G., Park, J.S., et al., 2016. Ultrasonography diagnosis and imaging-based management of thyroid nodules: revised korean society of thyroid radiology consensus statement and recommendations. Korean journal of radiology 17, 370–395.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

Tan, M., Le, Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR. pp. 6105–6114.

Tessler, F.N., Middleton, W.D., Grant, E.G., 2018. Thyroid imaging reporting and data system (ti-rads): a user's guide. Radiology 287, 29–36.

Townsend, J.T., 1971. Theoretical analysis of an alphabetic confusion matrix. Perception & Psychophysics 9, 40–50.

Vakili, M., Ghamsari, M., Rezaei, M., 2020. Performance analysis and comparison of machine and deep learning algorithms for iot data classification. arXiv preprint arXiv:2001.09636 .

Woo, S., Park, J., Lee, J.Y., Kweon, I.S., 2018. Cbam: Convolutional block attention module, in: Proceedings of the European conference on computer vision (ECCV), pp. 3–19.

Wu, H., Deng, Z., Zhang, B., Liu, Q., Chen, J., 2016. Classifier model based on machine learning algorithms: application to differential diagnosis of suspicious thyroid nodules via sonography. American Journal of Roentgenology 207, 859–864.

Zeiler, M.D., Fergus, R., 2014. Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer. pp. 818–833.

19
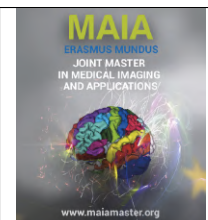
A Fully Automatic Algorithm for Scoliosis Assessment. Towards a clinical implementation

Francisco Aarón Tovar Sáez[a], Eva Vandersmissen[b]

[a]franciscoaaron.tovarsaez@agfa.com
[b]eva.vandersmissen@agfa.com

Abstract

Current clinical assessment of scoliosis, a lateral deformation of the spine, involves manual measurements of spinal x-rays from experts to guide intervention. Cobb angles are the gold standard in the assessment of scoliosis, measures as the angle between vertebrae. However, manual measurement of Cobb angles are time consuming and are subject to high inter-observer variability. Recently, many Deep Learning algorithms have been developed to automatically assess scoliosis but lack the potential to be implemented in clinical practice, mainly due to limitations in public datasets for scoliosis. In this work, a model designed with the focus to be implemented in clinical practice is developed consisting of three main blocks. The first one is composed by a Feature Pyramid Network with an EfficientNetB7 backbone to perform vertebrae segmentation. The second part of the proposed algorithm is to fit the endplates to the previous vertebrae masks using a connected components analysis and least squares fit method. Finally, the Cobb angles are measured following the current clinical practice. The algorithm predicts accurately the vertebrae masks and endplates fitting. The Cobb angles variability of the proposed method of the proposed method is 2.22º, considerably reducing the inter-observer variability found in clinical experts (3º-10º). Moreover, the generalization potential of the method is exploited by adding a generalization network to the algorithm, expanding its use in very different x-rays datasets. Visual assessment of the Cobb angles was performed in 4 different testing datasets, and feedback received from clinicians highlight the potential of the method to be implemented in clinical practice.

Keywords: Scoliosis, Cobb angle, Deep Learning, Vertebra segmentation

1. Introduction

The spine is the central bone of the human body to which all other bones are connected, participating in essential activities such as weight-bearing, movement or shock absorption. It is usually composed by 33 bones called vertebrae, divided in five regions along the spine. From top to bottom, the spine is divided in cervical (7 vertebrae), thoracic (12 vertebrae), lumbar (5 vertebrae), sacrum (5 vertebrae) and coccyx (4 vertebrae) regions. The upper 24 vertebrae are connected through intervertebral discs, articulated with high mobility whereas the lower nine are fused in adults (Kenneth, 2020).

Normal development and shape of the spine is crucial in the development of essential organs and other bones. The most prevalent disease related to the spine in children and adolescents is scoliosis, with a prevalence of 0.47%-5.20% (Konieczny et al. (2013)) . This spine condition is characterized by the deviation in coronal, sagittal and axial planes, named lateral curvature, thoracic lordosis (inward rounding of the back), and vertebral rotation (Kouwenhoven and Castelein (2008)). Instead of straight line, spines with scoliosis condition, usually have either C-shaped or S-shaped spines as shown in Figure 1.

The Scoliosis Research of Society defined scoliosis as a lateral deviation of minimum 10º in the spinal x-rays Bloch et al. (2012). Scoliosis is diagnosed as Idiopathic if no other condition is present such as congenital, neuromuscular or mesenchymal. Adolescent Idiopathic Scoliosis (AIS), accounts approximately for 90% of cases of idiopathic scoliosis in children between ages of 11 and 18 years (Konieczny et al. (2013)). Therefore, special focus is places on AIS in the devel-

Figure 1: C-shaped (a) and S-shaped (b) common scoliosis curves

Figure 2: Cobb angle measurement (Horng et al., 2019)

by measuring the angle between the same vertebrae used in previous studies to check the angle progression.

The spinal column vertebrae can be generally grouped, from top to bottom, into proximal thoracic (PT), main thoracic (MT) and thoracolumbar/lumbar (TL/L) regions. Therefore, complete assessment of scoliosis using the Cobb angle technique involves the calculation of 3 angles, representing the 3 curves in the spine (PT, MT and TL/L). While conventionally measured by protractor, Cobb angle measurement are nowadays calculated digitally mainly (Wills et al. (2007)). This process involves the manual location of each vertebra landmark on the relevant endplates and later automatic calculation of the Cobb angle.

However, Cobb angle measurement is time consuming and presents a high level of intra- and interobserver variability. This uncertainty is created by selection of different end vertebrae, drawing of the endplates and measurement of the angles (Gstoettner et al. (2007)). Usually, an interobserver variability between 3-5º is present in cases with mild scoliosis and can increase in cases of severe scoliosis up to 10º (Scholten and Veldhuizen (1987)). Moreover, the use of 2D x-rays inherently shows an incomplete view of the 3D nature of scoliosis and may yield an underestimation in the true Cobb angle of the patient. However, 2D x-rays remain the standard technique in clinical assessment of scoliosis.

2. State of the art

This section will review public datasets available for scoliosis assessment as well as recent works within the last three years involving Deep Learning techniques to asses scoliosis using 2D spinal x-rays. These methods will be broadly divided into two groups: landmark detection and image segmentation.

opment of clinical tools to automatically asses scoliosis.

Nearly 0.1% of scoliosis patients require surgery and around 10% require some kind of intervention ((Tambe et al., 2018)). Surgery is justified under the cases for which the lateral deviation is higher than 50º, since it may likely progress into the adult life, back pain, cardiopulmonary issues and psychosocial concerns. Surgical treatments include vertebral fusion, using screws and rods, and vertebral tethering aimed to stop complication rates and long-term consequences (Tambe et al. (2018)). Non-surgical treatments include scoliosis-specific exercise, or bracing.

Clinically, the diagnosis, treatment and follow-up of scoliosis is done using x-rays and a subsequent measurement called Cobb angle. The Cobb angle is nowadays the most common quantification used in scoliosis, originally proposed by the American orthopedic surgeon John Robert Cobb (Cobb (1948)) in 1948. Cobb angle is used to asses the degree of severity in scoliosis (as shown in Table 1).

Table 1: Scoliosis severity ranges and their Cobb angles

Severity	Cobb angle
Not scoliosis	<10º (Lau (2013))
Mild scoliosis	10º-30º (Bloch et al. (2012))
Moderate scoliosis	30-45º (Bloch et al. (2012))
Severe scoliosis	>45º (Bloch et al. (2012))

Cobb angle method involves the identification of the upper and lower endplates of the two most tilted vertebrae (shown in Fig 2). The angle between these endplates is denoted as the Cobb angle. The vertebrae involved in the Cobb angle measurement are highly important since they are used in future analysis for follow-up patients. Scoliosis in follow-up patients is assessed

2.1. Datasets for scoliosis assessment

In the past few years, there have been different challenges in computational methods and clinical applications for spine imaging (xVertSeg, IVDM3Seg and computational challenges on CSI). In MICCAI-CSI2014, for instance, two challenges were developed on "Spine and Vertebrae Segmentation" and "Vertebrae Localization and Identification". The commented challenges mainly focus on the analysis of vertebrae fracture and vertebrae localization in CT and MRI.

As part of the MICCAI-CSI2019, the first and the only (up to date) challenge was proposed for accurate automated spinal curvature (AASCE-2019 challenge). A total of 707 anterior-posterior (AP) x-ray spinal images for training and testing collected from London Health Sciences Center in Canada using EOS medical imaging system. These data was IRB approved. Training and testing datasets contain 609 and 98 x-ray images, respectively. Landmarks were provided by two professional doctors in London Health Sciences Center and are available publicly [1]. Since cervical vertebrae are rarely involved in spinal curvature (O'Brien et al. (2008)), 17 vertebrae (12 thoracic and 5 lumbar) are reported by specialists. Each vertebra is reported by four landmarks representing the four corners resulting in 68 points per image. The Cobb angles are also included as part of the dataset calculated from the landmarks.

Table 2 shows the relevant information of the AASCE challenge dataset for training images.

Table 2: Information of the AASCE dataset

Specification	Train
Number of images	609
Physical units (Dots Per Inch, DPI)	72
Pixels dimensions (SI units, micron)	144
Length of images (pixels) [min-max]	[973-3755]
Width of images (pixels) [min-max]	[355-1427]
Value of the Cobb angle (º) [min-max]	[0-56.39]
Number of not scoliosis (<10º)	20
Number of mild scoliosis (10º-30º)	190
Number of moderate scoliosis (30º-45º)	205
Number of severe scoliosis (>45º)	194

The main clinical limitation from this dataset is that the ground-truth Cobb angles are calculated using just one angle per vertebra (represented as the central line) instead of using the upper and lower endplates of the upper and lower vertebrae involved in the angle, as done in clinical practice. Figure 3 shows better the difference between calculating the Cobb angle using one line per vertebra (as done in the AASCE challenge) or using two endplates per vertebra (as done in clinical practice).
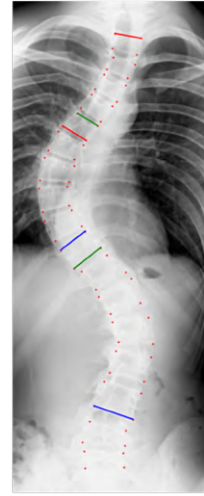
Figure 3: Two possible ways of measuring the PT, MT, TL/L Cobb angles. Left: using both upper and lower endplates for each vertebra, as done in clinical practice. Right: using one central line, as done in the AASCE challenge

2.2. Landmark detection algorithms

Boostnet (Wu et al. (2017)) is a neural network aimed to detect 68 landmarks (four corners of each vertebra) corresponding to 17 vertebrae (12 thoracic and 5 lumbar) in AP X-rays. Traditional ConvNet performance was improved by the addition of a BoostLayer, to remove outlier features, and a Spinal Structured Multi-Output Layer, to analyse spatial dependencies between different landmarks. These improvements yielded to a Pearson correlation coefficient of 0.94 between the ground-truth and the predicted landmarks. Even though the network predicted landmarks accurately, assessment of Cobb angle accuracy was not studied in this work. Data used in this study was part of the public database released for AASCE-2019 challenge.

Boostnet authors developed further the analysis of scoliosis assessment and published later the multi-view correlation network (MVC-Net) (Wu et al. (2018)) and multi-view extrapolation network (MVE-Net) (Chen et al. (2021)). These models measure Cobb angles using both coronal and sagittal 2D X-rays. In MVC-Net, authors predicted landmarks using a multi-view convolutional layer to exploit dependencies between both views in order to overcome the challenge caused by obstruction by the ribcage in the lateral view. In MVE-Net, landmarks are learnt using the previous multi-view convolutional layer as well as using each view independently. Each landmark prediction from these two methods are then combined using an inter-correction layer. Cobb angles were later calculated using the predicted landmarks. MVE-Net showed considerable better results than MVC-Net and Boostnet. The network was trained using 526 images (263 frontal and 263 lateral

x-rays) from a private dataset. Despite its promising results, these methods can only be applied if both frontal and lateral spinal x-rays are present for a patient, which is not the reality in clinical practice, since only frontal x-rays are acquired for most of the patients.

In Galbusera et al. (2019), a landmark detection network is applied to calculate landmarks not only to asses scoliosis but many other clinical parameters relevant in spinal disorders such as kyphosis, lordosis, pelvic incidence, sacral slope and pelvic tilt. A private database of 493 frontal and 493 lateral x-rays were used in this work. Even though predicted outputs strongly correlate with the ground-truth, the method suffers from a high standard error. Nevertheless, the model applies for a broad range of spinal disorders and in scoliosis specifically, other relevant clinical parameters such as the pelvic tilt or the trunk shift are also reported. Therefore, different clinical parameters should be considered in developing a clinical tool for scoliosis assessment.

2.3. Image segmentation algorithms

In Tan et al. (2018), authors used U-Net for image segmentation to classify each pixel in the image either as background, or vertebra. Ground-truth masks were obtained using a private dataset and due to data limitations, only the lumbar vertebra were considered in the study, therefore restricting the method to the lumbar region of the spine. From each individual vertebra mask, minimum bounding rectangle and least square methods were used to fit the upper and lower endplates of each vertebra. Predicted masks had a 98% accuracy with respect to the ground-truth masks. Cobb angle assessment was done using just one angle, as opposed to the full-spine PT, MT, TL/L standard Cobb angle measurement. A mean deviation of 1.7° was reported between predicted and manually calculated angles. Even though it showed promising results, the lack of analysis of the full spine segmentation restricts its potential implementation in clinical practice.

Similarly, in Horng et al. (2019) a U-Net is also applied for vertebral segmentation. However, instead of predicting all the vertebrae segmentation at once, U-Net is applied to segment each vertebra individually. From the original full frontal x-rays, spinal column is isolated from skull, limbs and hips using pixel intensity histograms in the horizontal and vertical directions. Each vertebra is then detected using polynomial fitting and histogram analysis. The area of each vertebra is the input to the neural network for segmentation. The work compared U-Net with Residual U-Net and Dense U-Net to perform the segmentation task. Then, minimum bounding rectangle was also used to fit endplates to each vertebra. However, the method was restricted to one endplate per vertebra (in the central part), as opposed to the clinical standard of two endplates per vertebra (upper and lower). Moreover, Cobb angle estimation was also limited to the maximum angle, instead of

the PT, MT, TL/L standard. The model was trained on 35 x-rays from a private database. The lack of data may impact in the ability of the network to generalize in real-world x-rays differing from the ones used in the private dataset.

DU-Net (Tu et al. (2019)) was proposed to segment the full spine mask instead of individual vertebra. The model combines an algorithm for spine detection with U-Net segmentation. 100 images were used for training and 10 for testing from a private database. The proposed DU-Net yielded better results in segmentation metrics than baseline U-Net. A 6th order polynomial was then fitted to the spine mask and Cobb angle was calculated using tangents to the curve. Again, a single angle was used to assess the Cobb angle calculation, as opposed to clinical practice, with the largest deviation from ground-truth angle of 5.4°. Even though its accurate results, this algorithm is far from being incorporated into clinical practice since vertebral endplates are not considered, restricting also its use to assess scoliosis in follow-up patients, as vertebrae involved in the Cobb angle should also be reported.

2.4. State of the art conclusion

Even though the previous analyses algorithms showed promising results in terms of accuracy in the tasks of landmark detection, image segmentation and eventually Cobb angle estimation, they are far from having the potential to be implemented in clinical practice. for scoliosis assessment. Landmark-based methods are extremely sensitive to noise. A small deviation in one of the predicted landmarks, may impact considerably the Cobb angle measurement. Moreover, the output of these methods are restricted to predicting the same number of vertebra. Therefore, its capabilities for generalization are limited, and a manual step of cropping the input images in cases with more vertebra is needed, which limits its potential to be fully automated. On the other hand, image segmentation techniques are more robust to small deviations from ground-truth and the influence in predicting a wrong Cobb angle is reduced. However, data is the main limitation, as no public dataset contains spine or vertebra segmentations. Moreover, private datasets used in image segmentation, are restricted in number of samples and field-of-view (only lumbar section, for instance).

Implementation in clinical practice of reviewed algorithms are also limited by the following reasons. Some algorithms rely on the spine segmentation, losing information about which vertebra are involved in the reported angles. Some other algorithms, predict directly the Cobb angles, losing the interpretability about the results, which usually is the main limitation for clinical implementation. Finally, most of the algorithms report just the maximum angle, instead of the clinical scoliosis assessment standard of PT, MT, TL/L Cobb angles.

Moreover, clinical experts consulted in this study suggest that measuring the Cobb angle using a central line per vertebra instead of upper and lower endplates, is not a valid method for clinical use. Therefore, all the works compared against the ground-truth Cobb angles of the AASCE dataset are not clinically valid.

2.5. Contribution of this work

To overcome the aforementioned pitfalls, we propose to the best of our knowledge, the first algorithm to assess scoliosis in a fully automatic manner with the potential to be implemented into clinical practice. The algorithm will be divided in three main parts to automatically report the PT, MT, TL/L Cobb angles in an interpretable way for radiologists, reporting not only the angles, but a visual guidance to highlight the vertebrae involved in each Cobb angle, allowing follow-up patients to be easily assessed.

To overcome the main limitation of landmark regression methods, the landmarks from the AASCE dataset are transformed into vertebra segmentations. Moreover, the ground-truth Cobb angles were recalculated, using the provided landmarks, considering two endplates for each vertebra (Figure 3 (a)), as opposed to the AASCE ground-truth Cobb angles, calculated using one line per vertebra (Figure 3 (b)).

Firstly, a Deep Learning model will be used to segment the vertebrae. Then, two endplates will be fitted for each segmented vertebra accounting for different angles per endplate. Finally, using the fitted endplates the three Cobb angles will be measured in the same way as it is done in clinical practice (Bloch et al. (2012)).

The proposed algorithm will not only be evaluated using the AASCE test dataset, but using three other private datasets to assess the generalization capabilities of the algorithm against different datasets. To account for the domain-shift problem, a novel approach will be considered.

3. Material and methods

The proposed framework for fully automatic assessment of scoliosis based on spinal AP x-rays consists of three phases: vertebra segmentation, endplates fitting and Cobb angle calculation.

3.1. Data pre-processing

AASCE ground-truth landmarks are converted to vertebra masks to avoid the main limitations of landmark regression algorithms.

3.1.1. Landmark errors correction

Some errors were present in the landmark coordinates of the AASCE challenge, including inconsistent ordering of the landmark coordinates, labelling of only

2 corners for some vertebrae and small spatial deviations from the vertebra corner. These errors may not be obvious when exploring the dataset but they suppose a limitation when designing an algorithm for creating the vertebra masks. About 10% of the training images were corrected for some errors. Therefore, any algorithm developed using the AASCE dataset will inherently contain errors associated with incorrect ground-truth landmarks, if not corrected.

3.1.2. From landmarks to vertebra masks

Figure 4 depicts the process used to convert the vertebrae landmarks to segmentations. Starting from the corrected landmarks (Figure 4 (a)), initial edges are considered as lines connecting the landmarks (Figure 4 (b)). For each of the 4 edges, vectors perpendicular to the edge are drawn for every pixel in the edge. Then, maximum change in intensity from bright to dark is stored in each perpendicular vector (Figure 4 (c)). Then, a 6th order polynomial is fitted for all the points in the edge (Figure 4 (d)). The process is repeated for every edge in the vertebra (upper, lower, left and right). Finally, all the fitted polynomials are connected using the boundary (MathWorks (2014)) MATLAB function. This function returns the boundary for a set of 2D points with a shrink factor parameter between 0 and 1. If 1, the function returns the Convex Hull transformation of the points and, if 0, it returns the compact boundary of the set of points.
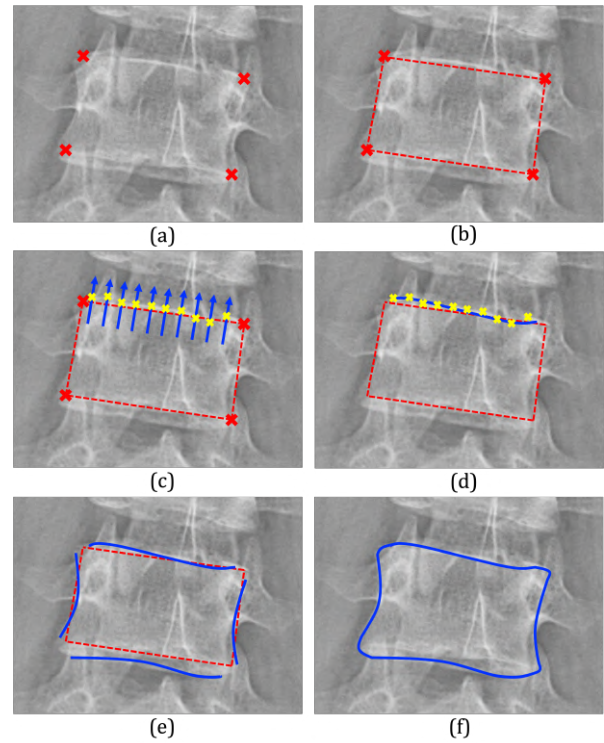

Figure 4: Process of converting AASCE dataset landmarks to vertebrae masks

3.2. (I) Vertebrae Segmentation

In this section, four of the most used Deep Learning models (U-Net (Ronneberger et al. (2015)), Linknet (Chaurasia and Culurciello (2017)), FPN (Feature Pyramid Network) (Lin et al. (2017)) and Pyramid Scene Parsing Network (PSPNet) (Zhao et al. (2017))). These models are combined with 12 different backbones in order to find the best combination of model-backbone for vertebra segmentation.

3.2.1. Data pre-processing

All the images present in the training set from the AASCE challenge (609 images with 20 non-scoliosis, 190 mild scoliosis, 205 moderate scoliosis and 194 severe scoliosis), were normalized in the range of (0, 1) and resized to a common size of 256x512 pixels. Moreover, the images were processed using contrast limited adaptive histogram equalization (CLAHE). Adaptive histogram equalization tends to overamplify noise in regions with similar contrast where the histogram is highly concentrated. In contrast, CLAHE limits the contrast amplification to reduce noise amplification.

This dataset was separated into training, validation and test sets using 70%, 15%, 15% of the dataset. This split was done so that all the sets are balanced in terms of number of images of different scoliosis types. To overcome the problem of limited number of samples in the training set, data augmentation was performed to randomly rotate the images with a random angle between -15º to 15º and random horizontal flip with probability of 0.5. Adding random Gaussian noise was explored but it did not improve the training result, therefore it was not considered in data augmentation.

Inspired by Xu et al. (2017), ground truth vertebra segmentations were added an extra channel: vertebrae masks edges. These masks were obtained by eroding the vertebrae masks with disk a structuring element of size 7 pixels and subtracting from the original vertebrae masks image. The idea of adding edges as another channel has been proven to bring the predicted segmentation closer to the true edges of the vertebrae. Finally, a third channel, the spine mask, was added to discard potential wrong segmentations in regions outside the spine where the model could identify vertebrae, such as the skull, limbs, or hips. The spine mask was calculated using the algorithm depicted in Figure 4 considering all the vertebrae as one single vertebra. Even though the prediction of the DL models will be a 3-channel output, only the vertebrae channel will be kept. The addition of the other channel was done to improve the prediction of the vertebrae masks channel. Figure 5 shows the pre-processed input image together with the 3 channels used as ground-truth to train the Deep Learning models.

3.2.2. Deep Learning model selection

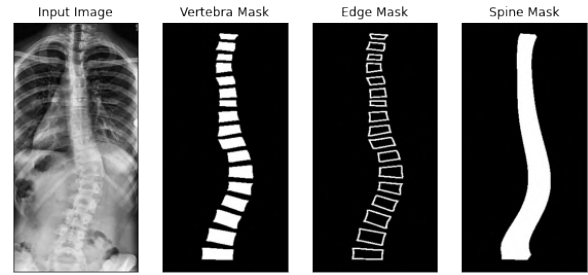In this work, four different models are used: U-Net, Linknet, FPN and PSPNet. Structurally, all the mod-

Figure 5: Modified AASCE dataset used for training the different DL models.

els architectures are similar, with an encoder-decoder shape where the encoder captures high-level hierarchical features from the input image while decreasing the matrix size of the image, and the decoder makes use of these features to create the final segmentation while recovering the original matrix size (Hu et al. (2019)). Therefore, it is the method of combining information from the encoder and decoder, the factor that differentiates the different models architecture. In case of U-Net, the encoder is used for multi-level feature extraction and the decoder combines these features with original encoder information through concatenations, using both features and spatial information into account. In the case of Linknet, modifies the basic U-Net architecture changing the fusion of high-level features and original encoder information. Instead of concatenation, the fusion technique is addition. PSPNet model creates a variable pooling layer from the lowest downsampled block of the encoder, resembling a pyramid. In this way, a vast collection of spatial resolutions are used to enrich the high-level features. Finally, FPN is similar to U-Net with the difference of applying 1x1 convolution layer to the encoder information and adding it to the different decoder layers. Finally, a double 3x3 convolution layer is applied to each block in the decoder and upsampled to the highest resolution. All the information is concatenated and a final 3x3 convolution is applied to get the output of the network. Figure 6 show visually the different architectures of the previously commented models.

12 backbones were selected as encoders for each of the four models. The backbones selected are VGG19, ResNet101, SE-ResNet101, ResNeXt101, SE-ResNeXt101, SENet154, DenseNet201, Inception-ResNetv2, MobileNetv2, EfficientNetB0, EfficientNetB3 and EfficientNetB7. The backbones were selected to account for a wide and broad range.

Architecture hyperparameters used in this work are present in Table 3.

Each experiment was trained for 70 epochs with a mini-batch size of 16 images. Adam optimisation was employed and the weights from the epoch yielding the lowest validation loss were saved. Learning rate was set to 0.001 with a scheduler function that reduces the
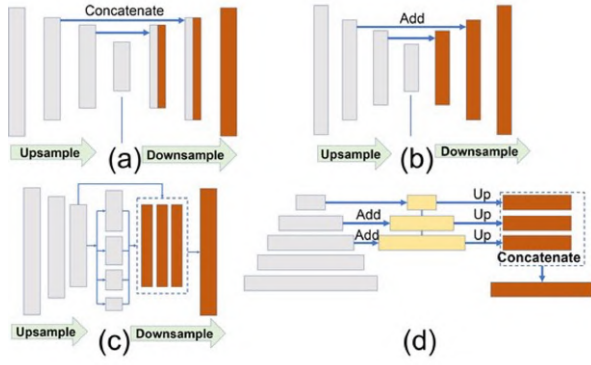
Figure 6: DL models used in this work. (a) U-Net, (b) Linknet, (c) PSPNet, (d) FPN. Parmar et al. (2020)

Table 3: Architecture hyper parameters for the different models

Model	Encoder depth	Batch Norm	Various
U-Net	5	Yes (dec)	filters per block= (16, 32, 64, 128, 256)
Linknet	5	Yes (dec)	filters per block= (16, 32, 64, 128, 256)
FPN	5	Yes (enc+dec)	pyramid filters=256 segment filters=128
PSPNet	5	No	output filters=512

learning rate by a factor of 10 after 10 consecutive epochs with no decrease in validation loss. For each backbone, weights were initialized using weights from trained models with ImageNet. Each model was trained using the Keras Deep Learning framework with Tensorflow backend using a Tesla V100 GPU.

DL experiment 1. On the choice of loss function. To choose the most appropriate loss function to train the models, a basic U-Net with hyperparameters shown in Table 3 was used with no backbone and random weight initialization. For this experiment, three different losses were considered: focal loss, dice loss, Tversky loss (Salehi et al. (2017)) and a combination of Tversky and focal loss, where focal loss is weighted by a factor β.

Focal loss is defined as

$$FL(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t)$$

where y represents the ground-truth label, p represents the predicted probability,

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{if } y = 0 \end{cases}$$

and

$$\alpha_t = \begin{cases} \alpha & \text{if } y = 1 \\ 1 - \alpha & \text{if } y = 0 \end{cases}$$

Focal loss is a function derived from the cross-entropy with two additional parameters (α and γ). α handles the class imbalance problem, and γ helps the

training focusing on misclassified pixels. Empirically, they are set to 0.25 and 0.2, respectively.

Dice loss is defined as

$$DL(y, \hat{y}) = 1 - \frac{2y\hat{y} + 1}{y + \hat{y} + 1}$$

where y represents the ground-truth label, $\hat{y}$ represents the predicted label and 1 is an added factor to account for edge cases where $\hat{y} = 0$ and $y = 0$.

Tversky loss is a generalization of Dice loss defined as

$$TL(y, \hat{y}) = \frac{y\hat{y}}{y\hat{y} + \beta(1 - y)\hat{y} + (1 - \beta)y(1 - \hat{y})}$$

where y represents the ground-truth label and $\hat{y}$ represents the predicted label. This loss weights differently FP (false positives) and FN (false negatives) with the help of the β coefficient.

The baseline U-Net was trained using focal loss, dice loss, Tversky loss with different β parameter value (0.2, 0.5 and 0.7), and combined Tversky and focal losses with an optimal Tversky β factor obtained from the previous teste and a focal loss weighted β of (0.2, 0.5, 0.7 and 1).

DL experiment 2. On the choice of optimum model-backbone combination. Using the optimal loss function obtained from 3.2.2, each of the four models are trained with all the different backbones commented in 3.2.2.

DL experiment 3. On the choice of training hyperparameters. Using the optimal model-backbone combination obtained from 3.2.2, several comparisons will be made to explore the effect of different hyperparameter selection. The comparisons made will be using ImageNet weights initialization vs random weights initialization, constant learning rate vs scheduled learning rate and complete encoder/decoder training vs decoder training.

3.2.3. Data post-processing

Post-processing is applied to the vertebra masks channel from the optimal model output. This processing includes outlier removal from pixels mistakenly classified as vertebrae, including objects that are significantly smaller than other predicted vertebra and spatial outliers. Moreover, to separate vertebrae that are fused by the network prediction, a combination of erosion, watershed segmentation and dilation is used.

3.2.4. Validation of vertebra segmentation

In order to validate each of the DL experiments performed to select the optimal model-backbone network and training hyperparameters, both test metrics such as Dice and Jaccard indices as well as training evolution will be used to compare the different tests.

Once the optimal model is trained, vertebra masks will be isolated from the output of the model (since it also contains the vertebra mask edges and the spine mask). To compare the predicted vertebra masks to the generated ground-truth vertebra masks, both the Dice Coefficient (overlap between predicted and ground-truth masks) and the Balanced Accuracy Rate (average of the proportions classified correctly for each individual class) will be calculated. These metrics will be evaluated before and after the post-processing step to check its influence. Moreover, qualitative assessment of the predicted vertebral segmentation will be done.

3.3. (II) Endplates fitting

Figure 7 depicts the process of fitting the endplates to a vertebra segmentation. From the vertebrae segmentation obtained from the network output, connected component analysis is performed to label each individual vertebrae as a different component. Then, for each vertebra, the centroid and the angle of its principal axis is calculated (Figure 7 (b)). The vertebra is rotated the same angle around its centroid. A horizontal line is drawn across its centroid and the intersection with the rotated vertebra is measured as the width. Then, a search area is established whose right limit is the right vertebra edge - 15% of the vertebra width and the left limit is the left limit edge + 15% of the vertebra width. This shrink is done to avoid the influence of irregular lateral edges of the rotated vertebra (Figure 7 (c)). Across the search area and for every pixel, a perpendicular vector to the horizontal line is drawn and points belonging to either upper or lower edges are saved (Figure 7 (d)). For all the upper and lower points, the endplates for the rotated vertebra are fitted using least squares fit (Figure 7). Finally, the upper, lower endplates and the vertebra are rotated back to the original orientation (Figure 7 (f)).

An outlier analysis is performed after all the upper and lower endplates are fitted to reduce the influence of a mistakenly predicted endplate. Once identified, the endplate is replaced by a weighted average of the endplates of its neighbours.

3.3.1. Validation of endplates fitting

The proposed fully automatic segmentation may yield to a different number of vertebra, not the 17 present on the ground-truth AASCE dataset. Therefore, when the number of predicted and ground-truth vertebrae are different, predicted and ground-truth endplates are resized to a common size that yields to the minimum absolute difference. In this way, endplates are comparable.

The metrics chosen to compare ground-truth and predicted endplates are the Mean Absolute Difference (MAD) and the Pearson correlation coefficient. MAD
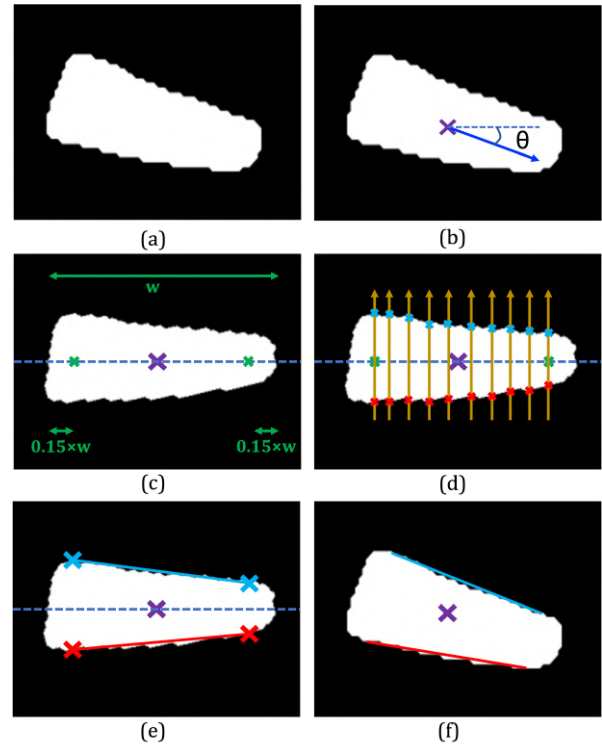

Figure 7: Process of fitting both upper and lower endplates from a segmented vertebrae

measures the magnitude of the expected deviation between both methods, while the Pearson correlation coefficient reflects the linear correlation between both methods. Apart from these quantitative metrics, other plots will be shown to analyse any possible bias in the predicted endplates.

3.4. (III) Cobb angle measurement

From the upper and lower endplates calculated from the previous part of the algorithm, the PT, MT and TL/L Cobb angles are measured automatically as done in clinical practice. The three curves are calculated as follows:

1. The apex of the MT curve is defined as the vertebra between T6 and T11 whose centroid is most horizontally deviated from the average centroid of vertebrae T1-T11.

2. The two vertebrae located upper and lower with respect to the MT apex that yield the highest angle are defined as the upper and lower vertebra of the MT curve, and the MT Cobb angle as this highest angle.

3. The upper vertebra of the MT curve is automatically defined as the lower vertebra of the PT curve. Then, the vertebra located upper with respect to this vertebra that yield to the biggest angle, is defined as the upper vertebra of the PT curve and this angle is defined as PT Cobb angle.

4. The lower vertebra of the MT curve is automatically defined as the upper vertebra of the TL/L curve. Then, the vertebra located lower with respect to this vertebra that yield to the biggest angle, is defined as the lower vertebra of the TL/L curve and this angle is defined as TL/L Cobb angle.

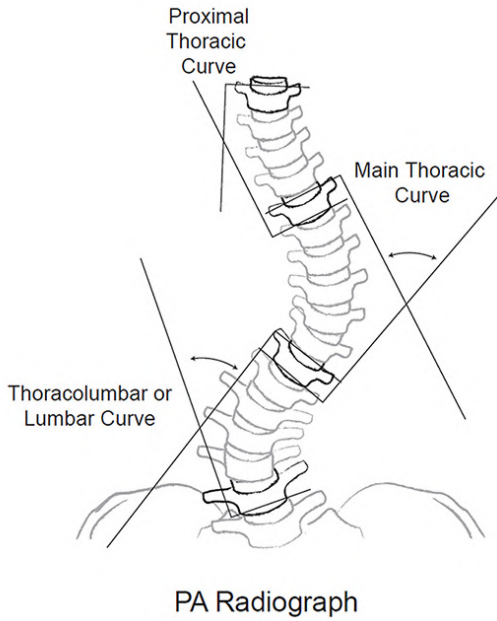Figure 8 shows a demonstration of these three Cobb angles.

Figure 8: PT, MT and TL/L Cobb angles (Group (2017))

3.4.1. Validation of Cobb angle measurement

The metrics chosen to compare ground-truth and predicted Cobb angles are the Mean Absolute Difference (MAD) and the Pearson correlation coefficient. Apart from these quantitative metrics, other plots will be shown to analyse any possible bias in the predicted endplates.

To better assess the inter-observer variability of Cobb angle measurement, a Saphiro-Wilk test is developed to assess the assumption of normality of the inter-observer difference. This normality analysis is later assessed using a Q-Q plot.

3.5. Generalization of the algorithm

One of the main limitations of the algorithms trained with the AASCE dataset is that the training data is cropped to only the spine region, restricting its use to other datasets where skull, limbs or pelvis are also present. This requires an added pre-processing manual step where the spine region should be cropped. This reflects a clear pitfall towards clinical implementation of scoliosis assessment algorithms, at least fully automatic ones. To overcome this limitation, this work introduces

a novel approach where first, a more robust segmentation is performed (lung segmentation) and using this mask, a spine ROI is automatically calculated and considered as the input for the vertebra segmentation network.

3.5.1. Dataset

A public dataset containing 800 AP x-rays with lungs segmentations obtained from the tuberculosis control program of the Department of Health and Human Services of Montgomery County, MD, USA (Candemir et al. (2013)).

3.5.2. Lung segmentation model

Baseline U-Net (same configuration as in 3.2.2 was used) was selected as the lung segmentation model, due to its simplicity and accurate results. Different backbones were analysed: SE-ResNeXt50, ResNet34 and DenseNet121. Data was separated into training, validation and testing using 70%, 15% and 15% rule, respectively. Training hyperparameters were selected as in 3.2.2.

Post-processing was applied to the output of the lung segmentation network. Connected component analysis was applied to keep only the two components with highest area.

3.5.3. Cropping of input image

Using the lung segmentation, a minimum bounding rectangle function is applied to both lungs mask. Upper and lateral edges of the minimum bounding rectangle were considered as the upper and lateral edges of the spine ROI. To calculate the lower edge of the spinal ROI, the height of the minimum bounding rectangle is multiplied by a factor of 1.7, accounting for the ratio between spine height and lungs height. This value was set empirically.

3.5.4. Final vertebra segmentation

The spine ROI is passed to the vertebrae segmentation network. An empty array of the same size as the original image is created and the vertebrae segmentation is placed in the spinal ROI, in this way reverting the cropping. Figure 9 show the overall process to segment the vertebrae in cases where the field-of-view is not the same as in the AASCE dataset.

3.5.5. Validation of the generalization of the algorithm

Trained lung segmentation models will be compared using the Dice Score Coefficient. Since the generalization algorithm is aimed to expand the vertebrae segmentation network to other datasets where no ground-truth is present, no quantitative measurements can be computed. Therefore, only visual inspection on the quality of the algorithm will be assessed.
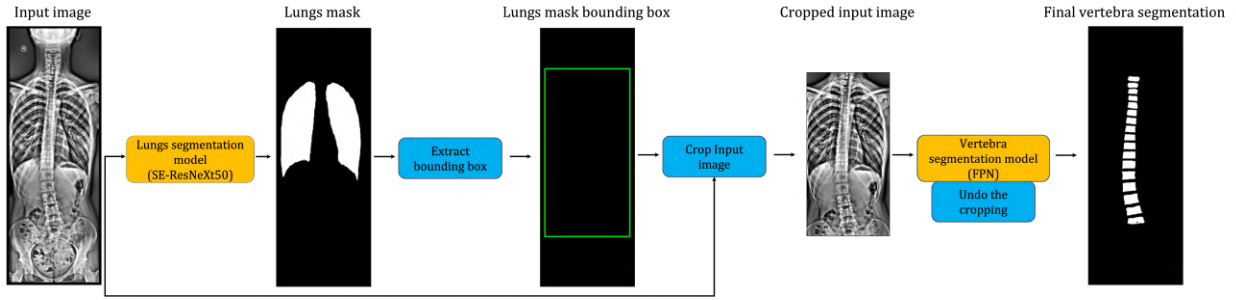
Figure 9: Process for segmenting the vertebra in different datasets with an increased field-of-view

4. Results

4.1. (I) Vertebrae segmentation

4.1.1. DL experiment 1. On the choice of loss function

Figure 10 shows the training evolution assessed using the validation dice score for different losses.
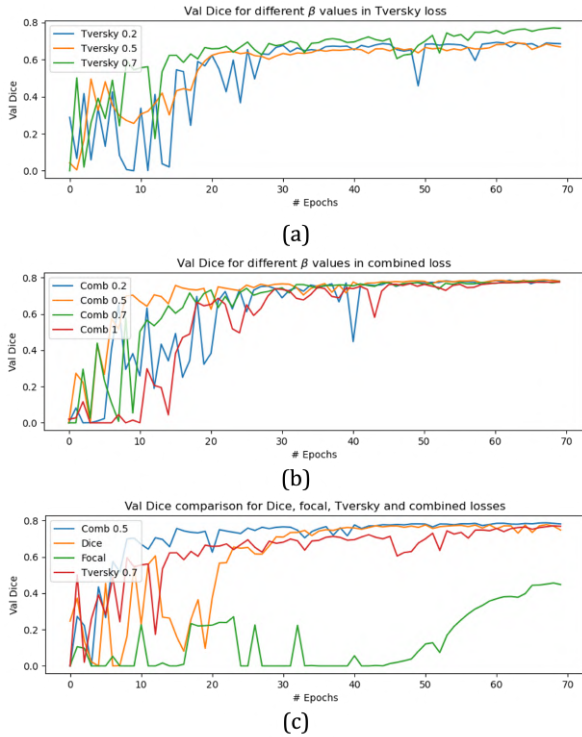

Figure 10: Training evolution assessed using the validation dice score for different losses

Figure 10 (a) shows the training evolution using different values of β for the Tversky loss. A value of 0.7 shows slightly faster training convergence and a final validation Dice score considerably higher than the rest of the cases. Values of 0.2 and 0.5 show similar behaviour in terms of convergence and final validation Dice score. Taking into account the imbalance of vertebra pixels with respect to background pixels (around 10% and 90% approximately), it is expected that a higher value of β will yield to better results.

Figure 10 (b) shows the training evolution using different values of β for the combined loss using Tversky loss (with a β value of 0.7) and focal loss. A value of 0.5 yields the fastest convergence, while 0.2 yields the slower training. In terms of performance, all the values yield to similar final validation Dice score.

Figure 10 (c) shows the training evolution using Tversky loss (with a β value of 0.7), focal loss, Dice loss and combined loss (with a β value of 0.5). Optimized Tversky and combined losses yield to both fastest convergence and higher final validation Dice score. Focal loss is the worst loss in terms of convergence and final validation metric. Since focal loss is a distribution-based loss, different segmentations could yield to similar distribution, so it is not suitable for segmentation tasks on its own. The optimal loss is selected as the combined loss between Tversky (β of 0.7) and focal loss, weighted by a factor of 0.5. Since focal loss is a distribution-based loss and Tversky a region-based loss, combination of both optimize both distribution and overlap simultaneously, yielding to the optimal loss function.

4.1.2. DL experiment 2. On the choice of optimum model-backbone combination

Figure 11 show the Dice and Jaccard indices for each of the four main models (U-Net, Linknet, PSPNet and FPN) trained using 12 different backbones for the test set. U-Net performs the best of the four models in terms of metrics and standard deviation. FPN performs closely to U-Net in terms both of metrics ans robustness. PSPNet yields the worst results overall.
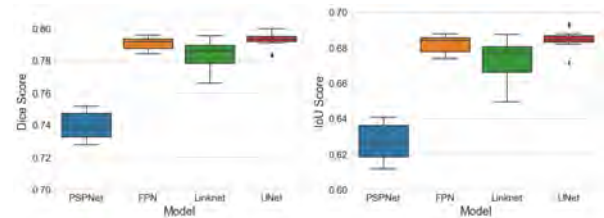

Figure 11: Dice scores (left) and Jaccard indices (right) for the test set grouped by models

Top-5 best backbone-model configurations test dice metrics are shown in Table 4. The top-5 best networks

perform extremely similar in terms of test dice, differing in just 0.005. Even though generally Linknet yields lower metrics than U-Net or FPN, it is the 4th best model when combined with EfficientNetB7 backbone. It is worth mentioning that not only FPN and U-Net models have a trend of better segmentation performance, but also the backbones. EfficientNetB7 is present in 3 of the top-5 best models.

Table 4: Top-5 model-backbones combination according to Test Dice

Model	Backbone	Test Dice
U-Net	SE-ResNext101	0.7999
FPN	DenseNet201	0.7957
U-Net	EfficientNetB7	0.7954
Linknet	EfficientNetB7	0.7944
FPN	EfficientNetB7	0.7941

Due to this similarity, choice of best model is difficult to perform. Therefore, visual inspection of some difficult test cases was performed to assess the optimal network. Figure 12 show one of these examples. From this comparison, it is clear that FPN and EfficientNetB7 yields the best vertebra segmentation in terms of not-merged vertebrae and regular vertebrae shape.
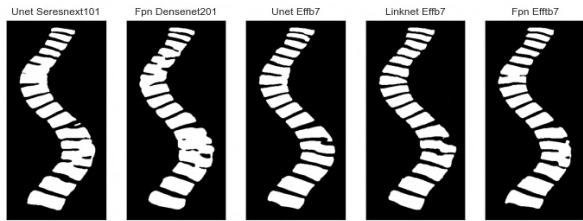

Figure 12: Visual comparison of the top-5 best model-backbone combination

From this comparison, the optimal model was defined as FPN with EfficientNetB7. Figure 21 shows a visual representation of the optimal network designed for vertebra segmentation.

4.1.3. DL experiment 3. On the choice of training hyperparameters

Figure 14 shows the training evolution assessed using the validation dice score for different training hyperparameter configurations.

Figure 14 (a) shows the training evolution assessed using the validation dice score for using static vs scheduled learning rate. Even though the convergence is very similar, the final validation Dice is slightly higher using a scheduler learning rate.

Figure 14 (b) shows the training evolution assessed using the validation dice score for training both the encoder-decoder or only the decoder. Training the full network achieved slightly faster convergence as well as improved final validation Dice score.

Figure 14 (c) shows the training evolution assessed using the validation dice score using ImageNet pretrained backbone weights, or random initialized weights . Using pre-trained weights yields a slight faster convergence and a better final validation Dice metric.

4.1.4. Vertebra segmentation results

Table 5 shows the Dice score and Balance accuracy rate between ground-truth and predicted vertebrae masks before and after post-processing. Post-processing yielded to a considerable increase of both metrics. These results prove that the post-processing designed in this work improves considerably the segmentation output. It is worth mentioning that these metrics were performed solely on the vertebra mask channel from the output of the network, as opposed to the previous, section, where the metrics were calculated for the 3 channels of the network output (vertebra masks, vertebra masks edges and spine mask).

Table 5: Vertebra segmentation metrics before and after postprocessing

Metric	Raw output	Post-processed output
Dice Score	0.8473	0.9130
Balanced Accuracy Rate	0.9273	0.9415

Visual assessment of the vertebrae segmentation step can be shown in Figure 15. A special remark should be placed in the fact that ground-truth vertebra masks are not real ground truth provided by experts, but a generated one from the landmarks. Therefore, there is an inherently error in the segmentation from the data used. However, Figure 15 shows how the prediction of the DL model actually adjusts much better to the true vertebra edges than the generated ground-truth.

4.2. (II) Endplates fitting

4.2.1. Quantitative analysis

Table 6 shows MAD and Pearson correlation coefficient comparing fitted endplates to ground-truth endplates.

Table 6: Metrics comparing fitted endplates vs ground-truth endplates

Metric	Value
Mean Absolute Difference	1.7315° ± 1.8079°
Pearson correlation coefficient	0.9817

Figure 16 (a) shows a strong linear correlation between predicted and GT endplates with no observable bias. The absence of bias in the prediction of endplates is also shown in Figure 16 (c), with a mean difference between the methods of 0.02° and a expected deviation using a 95% confidence interval of 0.02 ± 4.9. Figure 16
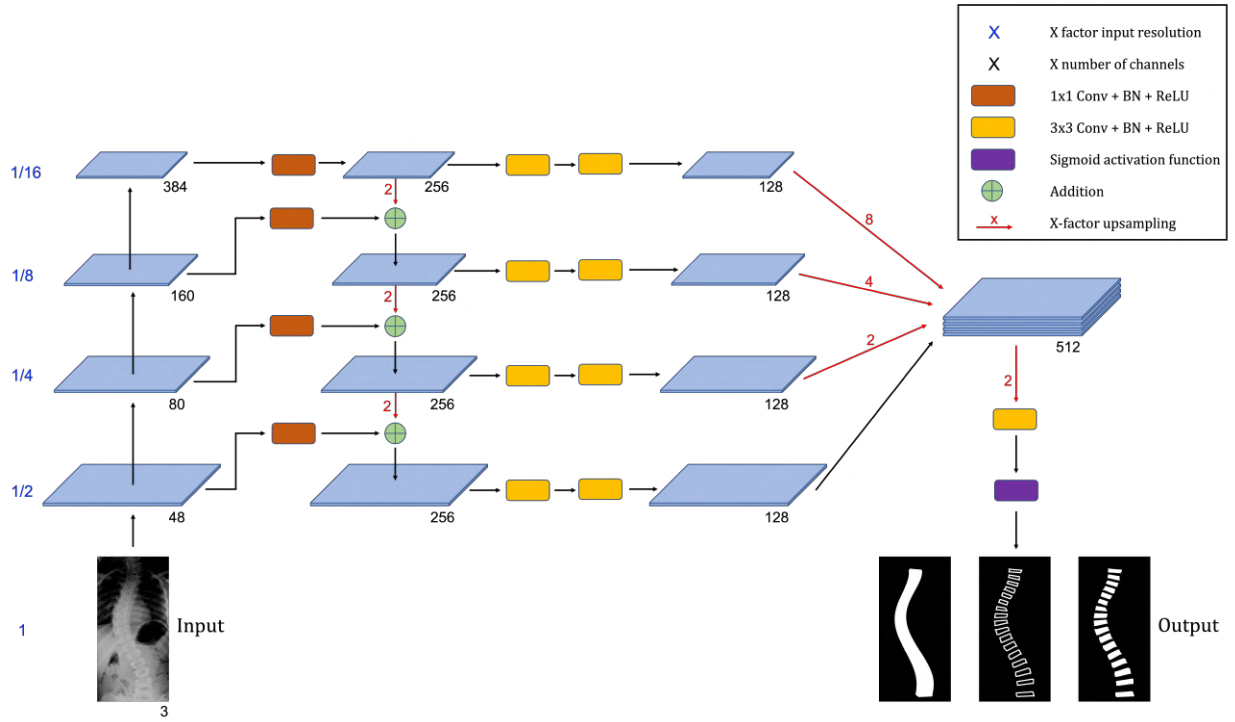
Figure 13: Optimal network (FPN model with EfficientNetB7 backbone) for vertebra segmentation
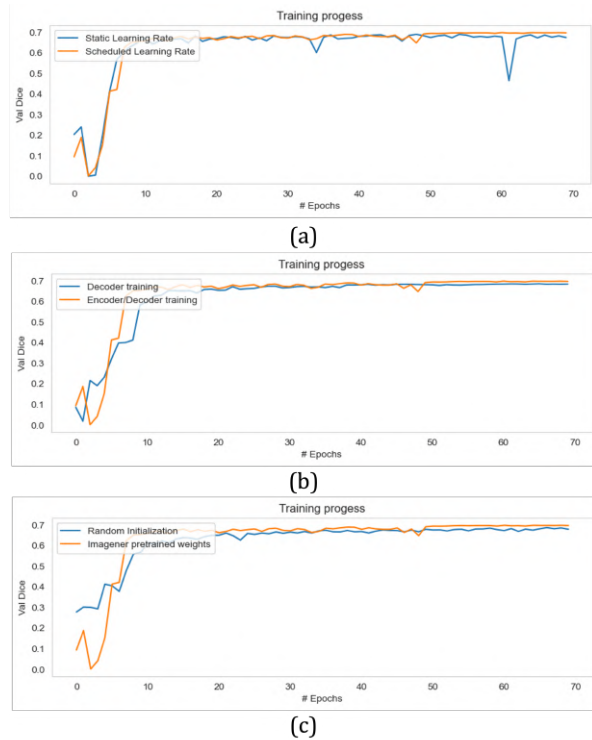

(a)

(b)

(c)

Figure 14: Training evolution assessed using the validation dice score for different training hyperparameters

(b) show the how close the difference distribution is to a normal distribution centered at 0 with a low variance, as desired.
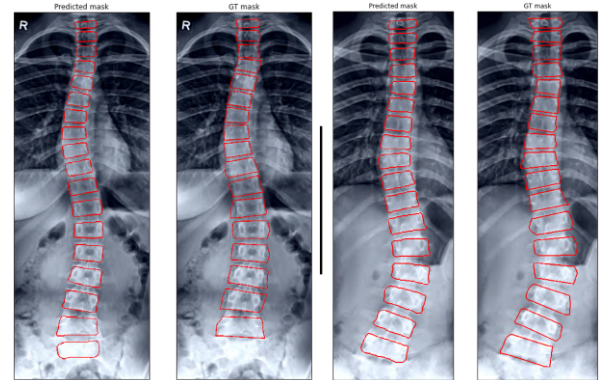

Figure 15: Visual assessment of the vertebrae segmentation prediction compared to the ground-truth one for two different cases

4.2.2. Qualitative analysis

Visual assessment of the endplates fitting step can be shown in Figure 17.

4.3. (III) Cobb angle measurement

4.3.1. Quantitative analysis

Table 7 shows MAD and Pearson correlation coefficient comparing predicted Cobb angles to ground-truth ones. It is worth mentioning that achieved variability (around 2º) is much lower than the inter-observer variability of experts (between 3º to 10º). However, as compared to endplates fitting, metrics indicate a slightly more variation. This is expected, as small variations of the fitted endplates could yield to bigger variations
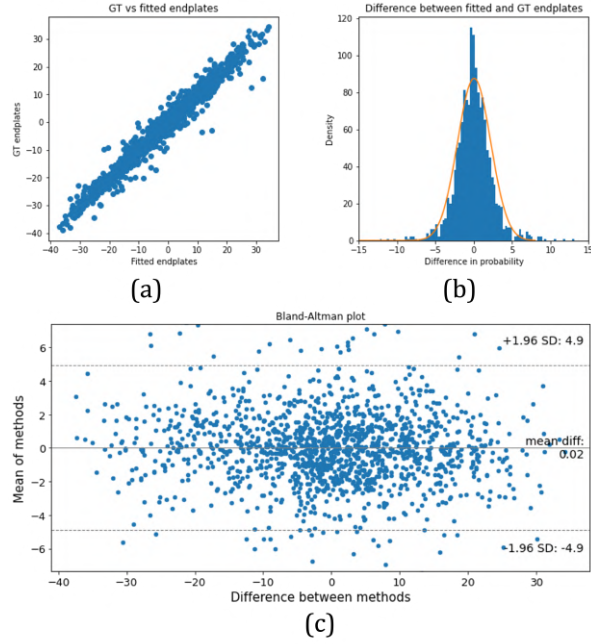
Figure 16: From (a) to (c), GT vs fitted endplates, difference between GT and fitted endplates and Bland-Altman plot
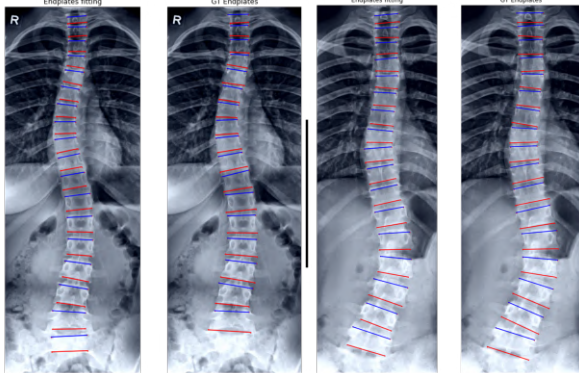

Figure 17: Visual assessment of the endplates fitted compared to the ground-truth ones for two different cases

in the Cobb angle measurement. It is also worth mentioning that this is not a real-world comparison, as the ground-truth angles were calculated from the landmarks and it would be desirable to compare against radiologists' manual measurement. This issue, limits to a certain degree the interpretation of this comparison. However, these results may yield an insight into the magnitude difference.

Table 7: Metrics comparing predicted vs ground-truth Cobb angles

Metric	Value
Mean Absolute Difference	2.2188º ± 2.0263º
Pearson correlation coefficient	0.9843

Figure 18 (a) shows a strong linear correlation between predicted and GT Cobb angles with no observ-

able bias. The absence of bias in the prediction of Cobb angles is also shown in Figure 18 (c), with a mean difference between the methods of 0.85º and a expected deviation using a 95% confidence interval of 0.85±5.65. Figure 18 (b) show the normality of the difference between both methods.
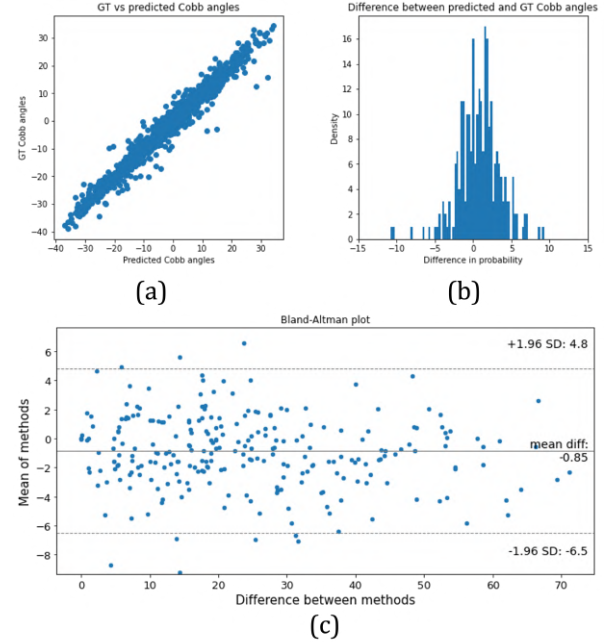

Figure 18: From (a) to (c), GT vs predicted Cobb angles, difference between GT and predicted Cobb angles and Bland-Altman plot

4.3.2. Normality in the inter-observer difference in Cobb angles

The Saphiro-walk test was applied to formally test the hypothesis that the difference between predicted and GT Cobb angles follows a normal distribution. The test concludes that the normality hypothesis should be rejected at the 95% confidence interval, as shown in Table 8 .

Table 8: Sapphire-Walk test for normality of the difference between predicted and GT Cobb angles

Metric	Value
W statistic	0.96
p-value	$1.18x10^{-5}$
Normal distribution hypothesis	False - rejected

Saphiro-Wilk test is very sensitive to small changes from normality, especially in the case with high number of samples. In this case, 3 angles per image in a total of 91 test images, yield to 273 data points in total, a relative high number. Therefore, further analysis to test the normality of the difference between predicted and GT Cobb angles was done. Figure 19 depicts a Q-Q plot to this purpose. A r^2 value of 0.960 indicates

that 96% of the variation in differences follows a normal distribution. Therefore, even though small changes from normality are present, the hypothesis of normality holds as true.
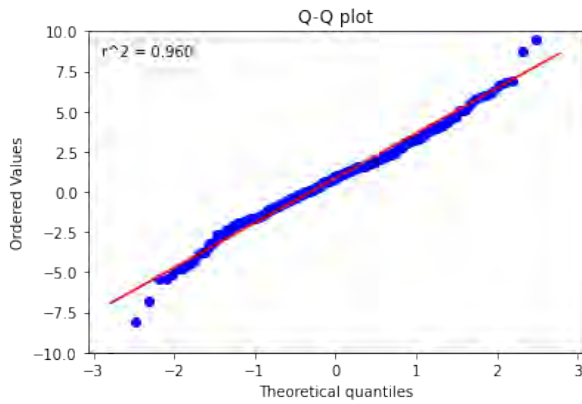

Figure 19: Q-Q plot of the difference between predicted and GT Cobb angles

4.3.3. Qualitative analysis

Visual assessment of the Cobb angles calculation step can be shown in Figure 20. Vertebrae involved in each Cobb angle are also reported, as well as the apical vertebra and labelling of the vertebrae.
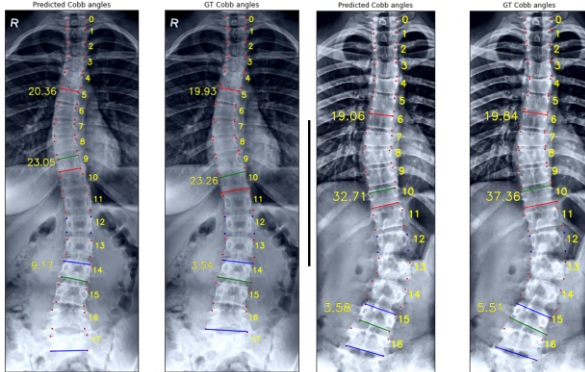

Figure 20: Visual assessment of the predicted Cobb angles compared to the ground-truth ones for two different cases

It can be noted that vertebrae segmentation network may output a different number of vertebra, yielding an inherent error present in the Cobb angle measurement. In the first case of Figure 20 (left), this effect can be observed. In this case, instead of segmenting a different number of vertebrae, the sacrum is wrongly detected as a vertebra, therefore, yielding to a different TL/L Cobb angle.

4.4. Generalization of the algorithm

4.4.1. Lung segmentation

Table 9 shows the different test dices for the three trained models for the lungs segmentation task. Best

metrics are reported for the SE-ResNeXt model. Therefore, this model is chosen to be the lungs segmentation network.

Table 9: Test dice for different models to segment the lungs

Model	Test Dice
SE-ResNeXt	0.8979
ResNet34	0.8712
DenseNet121	0.8725

4.4.2. Qualitative assessment

Figure 21 shows the whole automatic scoliosis assessment algorithm for different datasets. Row (a) shows a case from the real test set of the AASCE dataset. It is worth mentioning that, en though it was asked to the researchers that organize the challenge for the test set ground-truth, no response was obtained. Therefore, only visual assessment can be performed. Rows (b) - (d) show the full algorithm performance on different private datasets. It is worth mentioning that addition of the apical vertebra and label of the segmented vertebrae included in the visual output of the model, adds an extra value for the clinical implementation of the algorithm. Moreover, addition of the lung segmentation network, makes the follow-up of scoliosis patients possible, since the labels of the segmented vertebra will be consistent through different x-rays over time. This potential of clinical implementation was highlighted by different clinicians consulted in the development of this work.

5. Discussion and conclusion

The model presented in this work is as far as we know, the first potential attempt for fully automated scoliosis that could be implemented in clinical practice. First, an overview of the state-of-the-art algorithms was done to detect and highlight potential pitfalls in the clinical translation of these methods to the real-world. The proposed method solves these limitations, summarized as: converting the only publicly available dataset for curvature estimation to generate ground-truth vertebrae masks, redefining ground-truth Cobb angles taking into account the upper and lower endplates as advised by experts in the field consulted, addition of a generalization module to expand the model to new datasets and evaluation of the model in 4 different and diverse test sets. This generalization module could also be used to generate more training data far from the current AASCE dataset, improving the quality and size of future training data that could be used to train better segmentation models. In this work, no advanced DL model was explored, only U-Net, Linknet, FPN and PSPNet. While more advanced DL methods require bigger dataset, AASCE dataset contain just few hundred of images and these
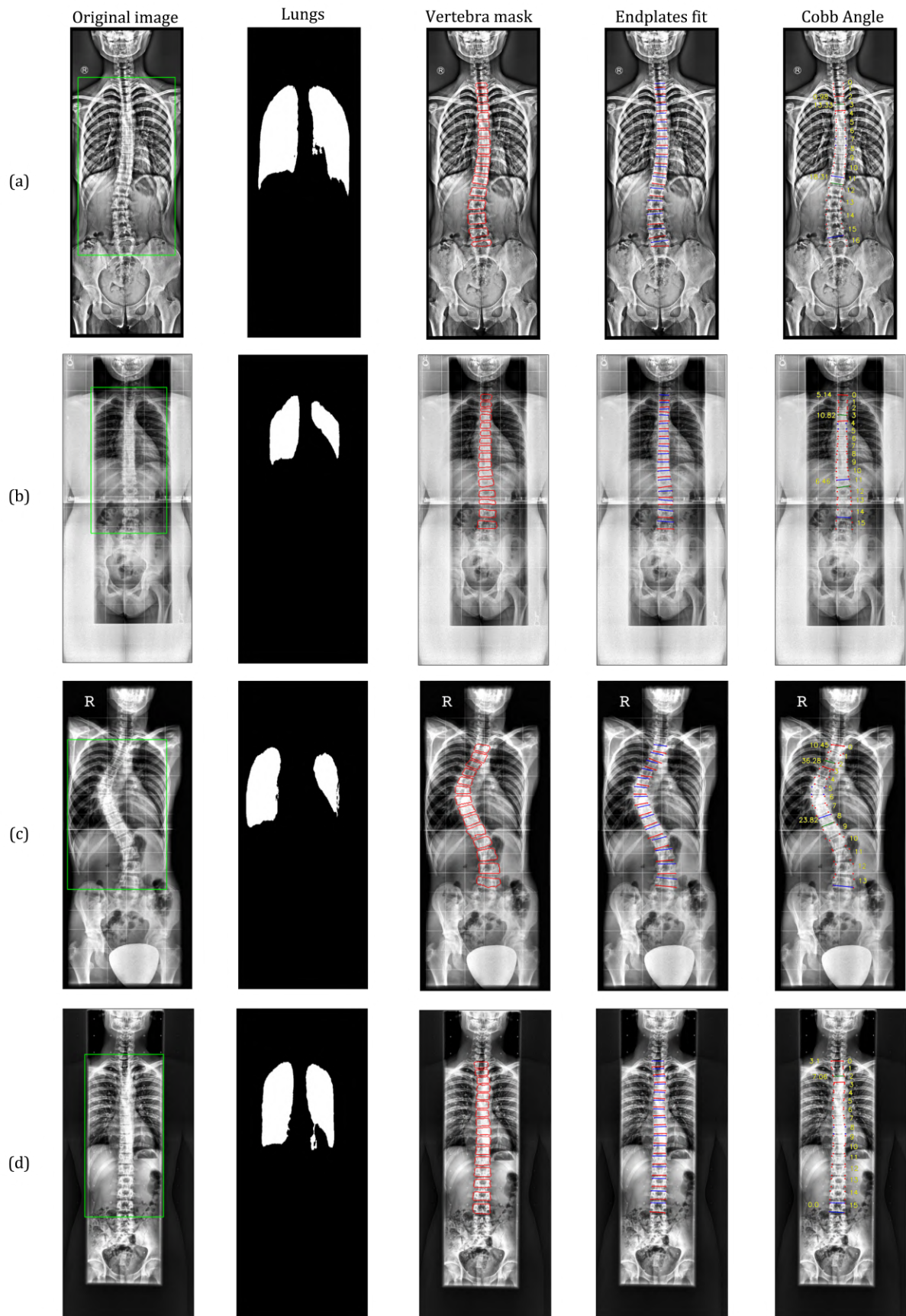
Figure 21: Optimal network (FPN model with EfficientNetB7 backbone) for vertebra segmentation

simple models have been proven to output highly accurate results.

The proposed model is presented in a modular algorithm, allowing for future development and improvement of individual blocks. Moreover, it restricts the DL part to segmentation uniquely, keeping the clinical explainaibility in the rest of the blocks, facilitating clinical implementation. The parts of the proposed algorithm are: vertebrae segmentation using a FPN model with an EfficientNetB7 backbone, endplates fitting and Cobb angle calculation. This work narrows considerably the gap between research and clinical reality. Up to know, feedback was received from different hospitals suggesting the potential of the algorithm for its clinical use. It is expected that a prototype could be used by different experts in the following weeks.

5.1. Limitations and future work

Even though the proposed method is claimed to be fully automatic, the generalization module contains a parameter that should be tuned for each patient: the multiplier factor of the lungs height to get the lower edge of the spine ROI. Empirically, this value was set to 1.7, working well with roughly 90% of the tested images. However, the ratio between spine height and lung height is dependent on each patient, so a model capable of tweaking automatically this parameter should be developed.

As it is designed, the proposed algorithm will mistakenly detect the apical vertebra for Cobb angle measurement in bending x-rays or images with a high pelvic tilt. Since it is calculated as the most horizontally deviated vertebra form the lumbar ones. in these cases, the apical vertebra will be considered as the top vertebra wrongly. Therefore, algorithms accounting for an analysis of the endplates evolution should be developed to automatically find correctly the apical vertebra and therefore, measuring the Cobb angle as in clinical practice.

With the proposed method, tools to automatically assess the scoliosis progression in patients could be easily developed, since vertebrae labelling is consistent through different x-rays, thanks to the lung segmentation module and a follow-up angle could be measured using the same vertebra as the previous examination. Moreover, a method to convert the vertebra labelling used (starting from 0, from top to bottom) into anatomical vertebrae labelling could be developed. Moreover, other clinical metrics relevant to the scoliosis disease can be incorporated into the last part of the algorithm such as trunk shift, pelvic incidence or pelvic tilt. The later could be used to automatically correct x-rays exams where the patient adopts a wrong position and the pelvis is tilted.

Finally, a clinical validation for Cobb angle measurement is still needed since the ground-truth data for the Cobb angles were generated. To this end, collaboration

with different scoliosis experts are being carried out and is expected to have some valuable feedback towards this clinical validity in the following weeks.
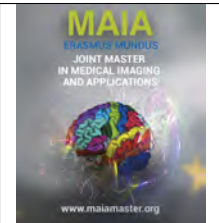
References

Bloch, K.E., Palange, P., Simonds, A., 2012. ERS Handbook: Self-assessment in respiratory medicine. European Respiratory Society.

Candemir, S., Jaeger, S., Palaniappan, K., Musco, J.P., Singh, R.K., Xue, Z., Karargyris, A., Antani, S., Thoma, G., McDonald, C.J., 2013. Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. IEEE transactions on medical imaging 33, 577–590.

Chaurasia, A., Culurciello, E., 2017. Linknet: Exploiting encoder representations for efficient semantic segmentation, in: 2017 IEEE Visual Communications and Image Processing (VCIP), IEEE. pp. 1–4.

Chen, X., Liu, J., Gong, C., Li, S., Pang, Y., Chen, B., 2021. Mve-net: An automatic 3-d structured mesh validity evaluation framework using deep neural networks. Computer-Aided Design 141, 103104.

Cobb, J., 1948. Outline for the study of scoliosis. Instr Course Lect AAOS 5, 261–275.

Galbusera, F., Niemeyer, F., Wilke, H.J., Bassani, T., Casaroli, G., Anania, C., Costa, F., Brayda-Bruno, M., Sconfienza, L.M., 2019. Fully automated radiological analysis of spinal disorders and deformities: a deep learning approach. European Spine Journal 28, 951–960.

Group, I.H.S., 2017. Lenke calculator. URL: https://hsg.settingscoliosisstraight.org/.

Gstoettner, M., Sekyra, K., Walochnik, N., Winter, P., Wachter, R., Bach, C.M., 2007. Inter-and intraobserver reliability assessment of the cobb angle: manual versus digital measurement tools. European Spine Journal 16, 1587–1592.

Horng, M.H., Kuok, C.P., Fu, M.J., Lin, C.J., Sun, Y.N., 2019. Cobb angle measurement of spine from x-ray images using convolutional neural network. Computational and mathematical methods in medicine 2019.

Hu, J., Li, L., Lin, Y., Wu, F., Zhao, J., 2019. A comparison and strategy of semantic segmentation on remote sensing images, in: The International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, Springer. pp. 21–29.

Kenneth, S.S., 2020. Anatomy & physiology: The unity of form and function. McGraw Hill.

Konieczny, M.R., Senyurt, H., Krauspe, R., 2013. Epidemiology of adolescent idiopathic scoliosis. Journal of children's orthopaedics 7, 3–9.

Kouwenhoven, J.W.M., Castelein, R.M., 2008. The pathogenesis of adolescent idiopathic scoliosis: review of the literature. Spine 33, 2898–2908.

Lau, K., 2013. The Complete Scoliosis Surgery Handbook for Patients: An In-Depth and Unbiased Look Into What to Expect Before and During Scoliosis Surgery. Health In Your Hands.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125.

MathWorks, 2014. Boundary of a set of points in 2d or 3d. URL: https://mathworks.com/help/matlab/boundary.html.

O'Brien, M., Kuklo, T., Blanke, K., Lenke, L., 2008. Radiographic measurement manual. spinal deformity study group (sdsg) medtronic sofamor danek usa.

Parmar, V., Bhatia, N., Negi, S., Suri, M., 2020. Exploration of optimized semantic segmentation architectures for edge-deployment on drones. arXiv preprint arXiv:2007.02839 .

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Salehi, S.S.M., Erdogmus, D., Gholipour, A., 2017. Tversky loss function for image segmentation using 3d fully convolutional deep networks, in: International workshop on machine learning in medical imaging, Springer. pp. 379–387.

Scholten, P., Veldhuizen, A., 1987. Analysis of cobb angle measurements in scoliosis. Clinical Biomechanics 2, 7–13. doi:https://doi.org/10.1016/0268-0033(87)90039-8.

Tambe, A., Panikkar, S., Millner, P., Tsirikos, A., 2018. Current concepts in the surgical management of adolescent idiopathic scoliosis. Bone Joint J 100, 415–424.

Tan, Z., Yang, K., Sun, Y., Wu, B., Tao, H., Hu, Y., Zhang, J., 2018. An automatic scoliosis diagnosis and measurement system based on deep learning, in: 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE. pp. 439–443.

Tu, Y., Wang, N., Tong, F., Chen, H., 2019. Automatic measurement algorithm of scoliosis cobb angle based on deep learning, in: Journal of Physics: Conference Series, IOP Publishing. p. 042100.

Wills, B.P., Auerbach, J.D., Zhu, X., Caird, M.S., Horn, B.D., Flynn, J.M., Drummond, D.S., Dormans, J.P., Ecker, M.L., 2007. Comparison of cobb angle measurement of scoliosis radiographs with preselected end vertebrae: Traditional: Versus: Digital acquisition. Spine 32, 98–105.

Wu, H., Bailey, C., Rasoulinejad, P., Li, S., 2017. Automatic landmark estimation for adolescent idiopathic scoliosis assessment using boostnet, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 127–135.

Wu, H., Bailey, C., Rasoulinejad, P., Li, S., 2018. Automated comprehensive adolescent idiopathic scoliosis assessment using mvc-net. Medical image analysis 48, 1–11.

Xu, Y., Li, Y., Wang, Y., Liu, M., Fan, Y., Lai, M., Eric, I., Chang, C., 2017. Gland instance segmentation using deep multichannel neural networks. IEEE Transactions on Biomedical Engineering 64, 2901–2912.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2881–2890.

Reflection Artifact Detection And Removal in Optoacoustic Imaging

Kudaibergen Urinbayev, Guillaume Zahnd

iThera Medical GmbH, Munich, Germany

Abstract

Optoacoustic imaging is a high optical contrast biomedical modality combined with high ultrsound spatial resolution in deep tissues. The contrast is created by pulsed laser beam directed into a tissue, where an image is reconstructed by a sensor data of induced ultrasound waves. One of the limitation of this technology is abundance of reflection artifacts that could cause an image misinterpretation and be an obstacle for correct quantification of functional information. Reflection artifacts appear when down-propagating ultrasound waves meet difference in acoustic impedance and reflect back. Thus, in optoacoustic images we could notice structures that are completely artificial.

In this work we introduce two novel approaches in reflection artifact detection and removal. First method is a segmentation deep learning approach trained on manually annotated artifacts. Overall, 166 anatomical regions for six participants were labelled, which correspond to a set of 4176 images. Due to varying borders and reflection appearance on different pulse wavelengths, the specificity is 0.949; We also obtained promising qualitative results. We found that network trained with weak supervision may over-perform the annotator.

In the second part, we generated pairs of sinograms with and without skin reflection artifacts. We trained a network that maps a sinogram containing an artifact to one without. We obtained almost perfect removal results on a synthetic test set; however, the results for a real in-vivo inference were poor. The domain adaptation was a problem for generalization.

Keywords: Optoacoustic imaging, reflection artifact, synthetic data, deep learning, ultrasound simulation, in-vivo imaging

1. Introduction

The history of optoacoustic (OA), also referred as photoacoustic, technology goes all the way back to Alexander Graham Bell. In 1880, he observed that when beam of light is projected onto different materials, they are, in return, capable to emit acoustic vibrations (Bell, 1880). He used this effect to create photophone - a telecommunication device that transmit acoustic messages through the light. It was also found that the same concept could be successfully used for a biomedical imaging purposes. Recently, research in this field emerged significantly.

In optoacoustic imaging infrared light, usually between 650 and 1200nm, is directed toward biological tissue, the tissue absorbs optical energy and increases in temperature, the resulting thermo-elastic expansion leads to ultrasound (US) emission. Generated ultra-sound waves are read with sensors in a transducer array, similarly to US imaging. In general, these modalities are closely related. Whereas for a contrast US is dependant on impedance differences (either caused by discrepancies in mechanical density or sound velocity), optoacoustic imaging relies on optical properties of a tissue. In addition, major difference is the magnitude of received ultrasound waves.

At different laser wavelengths chromophores (hemoglobin, melanin, lipids, water, etc) have different absorption spectrum. Therefore, with multiple pulses that last up to 100nm and mathematical unmixing of the received ultrasound waves at these wavelengths , it is possible to obtain functional information about the photoabsorbers (Ntziachristos and Razansky, 2010).

The application of the optoacoustic technology are extensive. It allows to quantify oxydized and de-oxidized hemoglobin in non-invasive manner, hence the

OA imaging could be used for cancer detection in the breast or skin imaging using handheld probe (Beard, 2011). OA imaging is also used for pharmacokinetics research in small animals. To generate a whole-body to-mographic image, usually a rodent is placed inside the machine and images are acquired live.

There are several challenges of OA imaging that pre-vent mass adaption of technology: unknown optical pa-rameters of biological structures, Cox et al. (2009), as well as occurrence of reflection artifacts. They are an artificial structures that do not correspond to any real tissues. The artifacts usually produced by bounced back ultrasound waves generated by photoabsorbers. It could lead to mistakes in quantification of functional measure-ments and misrepresentation of structural anatomy.

More precisely, underlying principle is showed in Fig. 2. First, an excitation pulse is emitted from the transducer (a). Once the laser beam reached an ab-sorber, the absorber undergoes photoacoustic effect (b) and generated ultrasound wave is propagated to all di-rection (c). US waves that were propagating up are read by the scanner (d). They represent the real ab-sorber in the reconstructed image. However, waves that were spreading deeper into the tissue meet difference in densities, which leads to impedance mismatch. This mismatch acts as a mirror and the reflected ultrasound waves move up and read by the scanner(e). Thus, the absorber appear second time on the reconstructed op-toacoustic image.

Acoustic impedance (Z) is a physical property that defined by resistance a tissue encountered by ultrasound wave propagation. The formula is following:

$$Z = \rho * c, \tag{1}$$

where Z stands for impedance, ρ is mechanical density and c is speed of sound. Therefore, if mechanical den-sity rises, so does impedance.
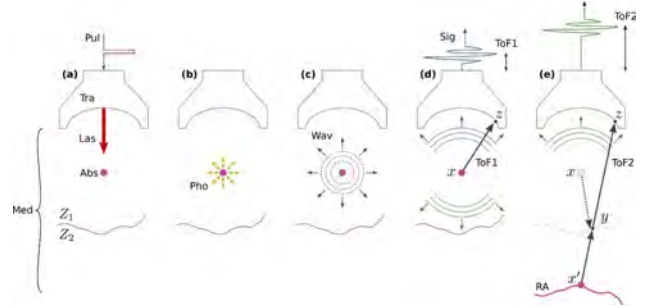

Figure 2: Physical principle of reflection artifact. Pul - pulse. Tra - transducer. Las - laser beam. Med - medium. Z_1 and Z_2 - acoustic impedance. Pho - photoacoustic effect. Wav - ultrasound wave. ToF - time of flight, RA - reflection artifact. Image credits: iThera Medical GmbH

The Fresnel equations describe the reflection and transmission light between two optical media (Fig. 3).

$$\begin{cases} R = \left(\dfrac{Z_2 cos(\theta_i) - Z_1 cos(\theta_t)}{Z_2 cos(\theta_i) + Z_1 cos(\theta_t)} \right)^2 \\ T = 1 - R, \end{cases} \tag{2}$$

where M_1 and M_2 represent two media, T and R are transmission and reflection coefficients, respectively. Z_1 and Z_2 are impedance. θ_i and θ_t are incident and trans-mission angles.

The artifact appear almost on every OA image with a different magnitude. Most frequent sources of reflection artifacts are skin and superficial vasculature. The exam-ple of the skin reflection could be seen in Fig. 1. In this work, we attempted two different methods to identify and tackle the artifacts. The first method is detection based on the segmentation of reflections, which were manually annotated by the optoacoustic specialist. In the second part, we tried to remove reflections in sig-nal domain based completely on synthetically generated data.

Figure 1: Reflection artifact example. Ultrasound image (a), OA image (b-c). Yellow arrows point to the skin reflection artifact. Red dotted line shows the photoabsorber that is the source of reflected ultrasound wave. Blue dashed line depicts the impedance mismatch that act as a reflector. Beige dashed line indicates the reflection artifact. For display purposes only, all of optoacoustic images are processed with CLAHE and ultrasound images were clipped between -20 and 20.
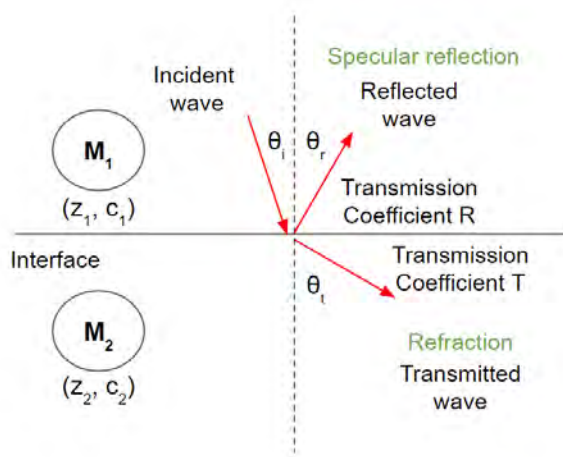
Figure 3: Refraction and specular reflection

2. State of the art

There were proposed many methods to tackle reflection artifacts. Jaeger et al. (2009) proposed a technique, where in a series of OA images, by manually applying pressure (palpation), it is possible to decorrelate reflections in deforming tissues. However, this approach works only in soft tissues and extensive training is still required. Petrosyan et al. (2018) suggests to use US focused beam to localize tissue vibration. Therefore, it is possible to differentiate the artifact by comparing images before and after deformation.

Photoacoustic-guided focused ultrasound (PAFU-Sion) is a method that uses focused US pulse acquisitions to replicate reflection artefacts in OA signal (Singh and Steenbergen, 2015). It is only feasible to identify reflections in focal zone of US beam and there is no real time application of the method.

Nguyen et al. (2018) suggested to identify reflection artifacts on multi-wavelengths OA images, taking into account the spectral response of an absorber and it's corresponding reflection. Later, the proposed the same approach, but utilizing only two wavelengths excitation Nguyen and Steenbergen (2020). These methods are not automatic and require prior knowledge of reflection source.

Deep learning based approach is also one of the methods that was used to combat the reflection artifacts. There are a limited number of papers, which focus on this topic. In general, below mentioned works are not generalizable and constrained to specific areas of interest, such as artifacts generated by point-like targets (needle tip) or focusing on OA images of human fingers.

Allman et al. (2018) used convolution neural networks to locate point-targets that produce reflection artifacts. In an in-vivo environment such point-like targets could be cross-sectional tips of the needle, catheters, or brachytherapy seeds. Training data was completely generated synthetically and the reflection itself was mimicked by shifting the real source wavefront deeper in signal domain. They implemented Region-Based Convolution Neural Network (R-CNN), an object detection deep learning algorithm, which is able to locate the region of interest. Thus, they were able to classify and detect a source and a reflection in the sinogram. There are some limitations to this approach. First of all, the work is restricted only to point-like hyperechoic structures and it could not be extended for frequently appearing skin reflections.

In their work, Shan et al. (2019), used convolution neural networks primarily to accelerate OA image reconstruction, as well as removing reflection artifacts. The main idea of the work was to map the first iteration of a reconstruction algorithm to the last one using deep learning, hence accelerating the process, skipping the mid-iterations. They assumed that reference algorithms remove reflection artifacts using numerical methods. But the reflections they were referring are cloud-like noise and it does not extend to the superficial structures like reflections of skin.

Agrawal et al. (2021) proposed a reflection removal approach that focuses on OA images of human fingers. They simulated a dataset of realistic digital phantoms of human fingers and applied a convolution neural network, U-net that mapped B-mode image with artifact to the reflectionless twin image. In order to obtain a training dataset they randomly generated anatomically plausible images of human fingers on a simulation software that they proposed earlier. After developing the U-net algorithm they qualitatively evaluated it using physical phantoms and in-vivo samples. The main limitation of this method is that it does not expand beyond a human finger.

3. Material and methods

3.1. Reflection artifact detection using annotated data

In this part of the project, we annotated reconstructed OA images for reflection artifacts. We developed a segmentation deep learning model using the annotated data.

3.1.1. Data for the annotation

For this part, data was taken from the work of Dehner et al. (2022). The dataset consists of acquired in-vivo images and synthetically generated manifold images and respective sinograms. All of the images were reconstructed using physical characteristics of MSOT Acuity Echo scanner, iThera Medical GmbH.

Overall, for in-vivo image set multiple anatomical regions were obtained, such as scans from biceps, carotid, etc. Those regions were also scanned in different probe
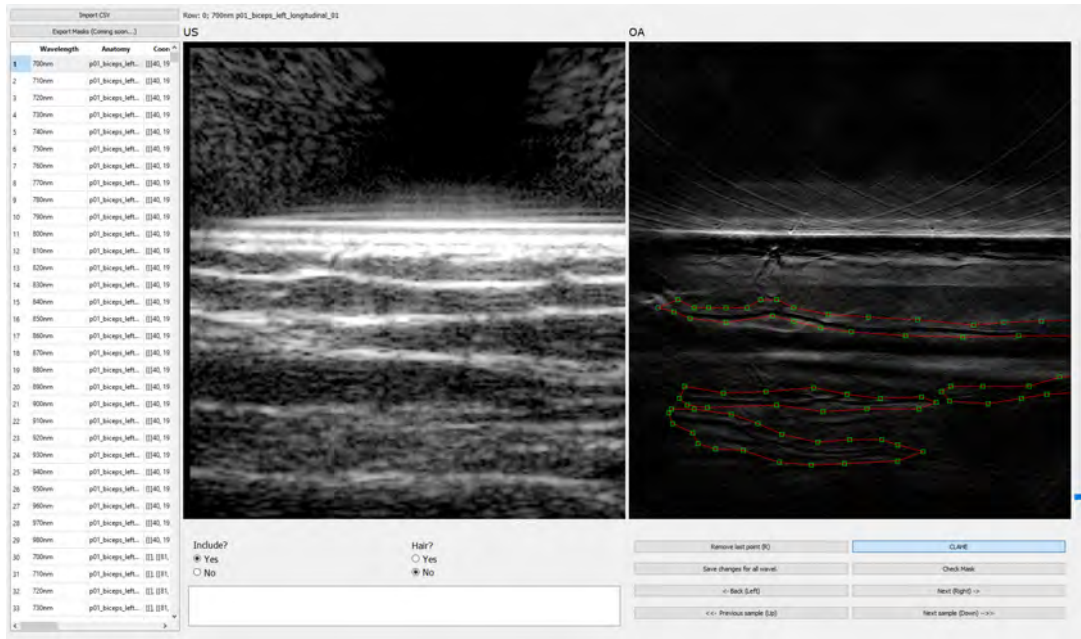
Figure 4: Developed user interface of annotation software, where reflections are enclosed in polygons on the OA image in the right window. On the left window a corresponding US image is shown for the reference

orientations. For each sample, 29 laser wavelengths ranging between 700nm and 980nm were examined. Acquired sinograms were band-passed and cropped to correct for device specific noise. They were reconstructed with iterative-model-based to 416 by 416px, which correspond to 4.16 by 4.16cm. In total, about 4800 in-vivo images were used.

3.1.2. Annotation

For the network training data, custom software was developed from scratch to minimize specialist's work on annotations. The software was written using PyQT5 library, a Python based Qt GUI framework, and PyQt-Graph plotting software that complements functionality of PyQT5.

The user interface is shown on the Fig. 4. Since reflections are created from the impedance mismatch, it was necessary to display US for reflection validation. Therefore, the screen was divided into two windows - ultrasound and optoacoustic image. For display purposes, image could be processed with contrast-limited adaptive histogram equalization (CLAHE) or the value of contrast could be clipped using a vertical slider on the most right of the interface. In order to enclose reflections, one should place a consecutive green handles using left click button to create a region of interest (ROI). It is possible to delete last handle or the whole ROI. The position of the handles is then saved in the CSV file that plays the role of back-end database. Data of the file is displayed on the left-most side. The table contains information about the participant's anatomy, wavelength, position of the handles, as well as paths of the images. While annotating, a specialist must point out whether

to include image to the training dataset or not, due to image corruption. The main reason of the corruption is the hair, they populate an image with numerous reflection artifacts. Therefore, the hair checkbox was also included. The annotator could leave a comment in the white box below that will be registered in the CSV file as well.

The annotation process was done by a single specialist in optoacoustic imaging with an experience of more than five years. The annotation was focused on the skin generated artifacts; however, several other structures, like probe membrane and hair reflections, were bounded as well. ROIs were annotated only on each image with wavelength of 700 nm.

For each anatomical region, there are 29 self-registered images of wavelengths 700 to 980 nm. Due to optical absorption, tissues have different appearance on an reconstructed image that depends on corresponding pulse wavelength. Reflection artifacts may also have different manifestations depending on the absorption spectrum of the source. For skin reflections, the source is usually a melanin, which is contained in epidermis. The comparison of reflection artifacts on different wavelengths is depicted on the Fig. 6.

Thus, 144 unique annotations were done. Out of these 144 anatomical regions 16 did not had any labeled reflections, hence in this data set almost 90% of images contain reflections to some extent. For each annotation there was 29 images of different wavelengths with overall of 4176 images. Data per participant is shown in Table 1. The process took 3 hours.

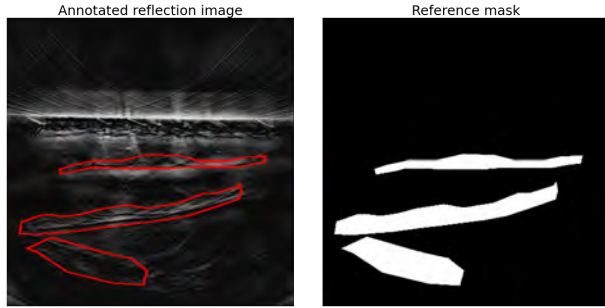After a qualitative analysis of the annotation, we re-

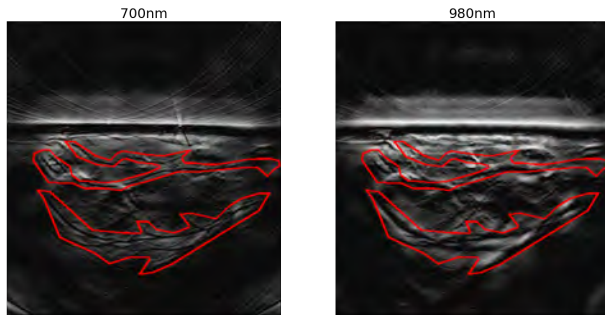Figure 5: Conversion from polygon annotation to mask

Figure 6: Comparison of OA images of same anatomical region with different wavelength spectrum: 700 and 980nm. In low wavelengths reflection are crisp, while on high wavelength reflections are blurred and wide. Red polygons indicate bounded reflection artifacts

Participant	Number of anatomical region	Total number of images with a reference
01	17	493
02	20	580
03	29	841
04	27	783
05	32	928
06	19	551

Table 1: Annotated data description
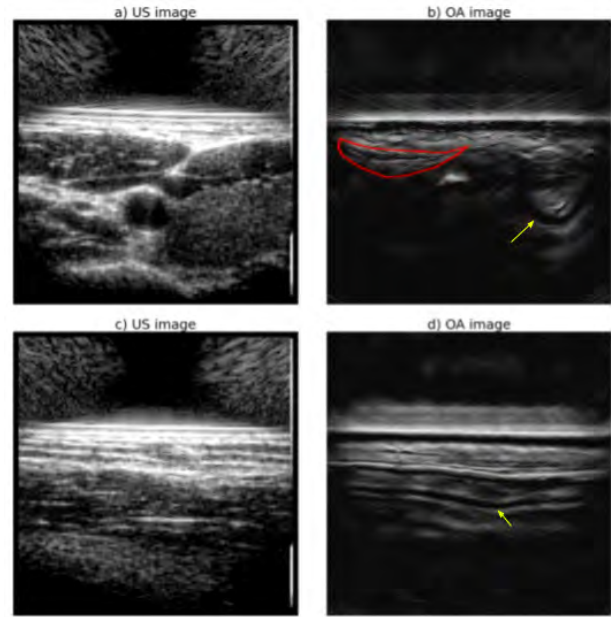
Figure 7: Mislabel of artifacts. Yellow arrows indicate missed reflections during annotation

alized that significant number of reflections were not labeled. Since most of the images include reflection artifact to different magnitude and scale, it was possible that the optoacoustic specialist missed them. Hence, we believe that obtained dataset was weakly labeled. The examples of mislabeled data are seen in Fig. 7.

The end goal of the project is to segment the outline of the reflections, the polygons were filled and converted to binary masks. Samples with no labeled reflections were converted to uniform image with intensity value of 0. The example is shown in Fig. 5.

3.1.3. Deep learning

Our main objective was to detect reflections on the image. For this purpose we decided to pursue with segmentation task.

After several iterations, Unet model with Resnet34 as an encoder was chosen. The implementation of this model was used from the "segmentation_models" library for PyTorch (Yakubovskiy, 2020). For the model bias term was enabled; kernel size was set to 5×5 with 2×2 padding. Encoder weights were randomly initiated. Decoder depth was set to 3 layers and number of filters on the first one was 128. The activation function for the output was set to sigmoid.

In OA images skin is usually hyper-intense. To account for this high intensity images were clipped between 0 and 255 and later normalized between 0 and 1.

Moreover, due to limited sample size several augmentation techniques were used with the help of Albumentation library developed by Buslaev et al. (2020). It was necessary to preserve the structure of the reflections, hence no warpping techniques were used. Hence, HorizontalFlip and ShiftScaleRotate (shift limit of ±0.05 and rotation limit of ±20) augmentations with the same probability of 0.5 were picked.

For the optimizer, Adam was chosen with Learning rate of 0.001 and LR scheduler with step size of 1, as well as gamma of 0.99. For the loss we chose Mean Squared Error or MSE.

For the reported results cross validation was used, where six participant permutations of 4-1-1 train-val-test split was carried out. The test scores were shown from the epoch with the highest Dice score on validation set.

3.1.4. Evaluation metrics

We evaluated segmentation results using different metrics:

$$\text{Specificity} = \frac{TN}{TN + FP}, \qquad (3)$$

$$\text{Recall (Sensitivity)} = \frac{TP}{TP + FN}, \qquad (4)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad (5)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \qquad (6)$$

$$\text{Dice score} = \frac{2TP}{2TP + FP + FN}, \qquad (7)$$

where TN is true negative, TP is true positive, FN is false negative, and FP is false positive. Specificity is a percentage of negative values, which were identified correctly. Recall or sensitivity is a percentage of positive values, which were identified correctly. Precision is a measure of the quality of a positive prediction. Accuracy is a percentage of correctly predicted labels and dice score is a measure of similarity.

3.2. Synthetic data-based artifact removal

In this project, which is separate from the reflection annotation project mentioned in section 3.1, we designed a pipeline, where we would remove skin reflection using a deep learning algorithm trained completely on simulated data in signal domain.

We decided to remove artifacts in sinogram domain, because there are multiple number of image reconstruction algorithms that are used both in production and research. Therefore, it will be convenient to use the artifact removal model for several parties.

There are 3 main steps in this project: 1) Using simulation software we created a dataset which contains pairs of images: sinograms (signal channels) with the artifact and ones without but with exactly the same background content. 2) We developed an algorithm that learns to map sinogram containing the artifact to one that does not have reflections. Thus, the algorithm is able to learn and produce the artifact-free image if it would be given a new sinogram with a reflection. 3) The most important step of the development is the evaluation of the algorithm. The in-vivo data is taken from section 3.1.

3.3. Simulation setup

The main objective is to obtain a sinogram with physically realistic reflection artifact in it. k-Wave software was chosen for this task. K-Wave is an open source, third party, Matlab toolbox developed to simulate the acoustic waves (Treeby and Cox, 2010). It was chosen because it allows to generate reflection artifacts by manipulation of mechanical density. Since reflections, e.g.

in in-vivo OA images, are produced by sudden change of impedance between tissues, this is a crucial functionality that is absent in several other simulation software.

Thus, in K-wave it is possible to setup acoustic pressure field that is developed after the photoacoustic effect. In order to be more generalizable we decided to use general images that depict real-world objects and patterns from The PASCAL Visual Object Classes Challenge 2012 (VOC2012) (Everingham et al., 2012). This way, we try to capture diverse feature representations for background. However, in order to fit the domain of in-vivo images, it is necessary to use the machine model-based operator (Rosenthal et al., 2014). We took these processed images from DeepMB dataset (Dehner et al., 2022).

To this image we added a modelled structure that anatomically resembles a skin. We generated a random line using 4 component FFT and placed it over the image for initial pressure. The inserted skin exceeds the maximum pressure value of the background image by 1.5-4 folds. The width of the line varies between 1 to 4 pixels. A Gaussian filter with random standard deviation was also applied to the synthetic skin line.

To generate a reflection artifact, impedance mismatch should be designed. For this, we set binary mechanical density over the medium grid with random phase. The density mismatch line was modelled similarly to skin line and placed deeper, with respect to the transducer, than skin line. An example could be seen in Fig. 11. For the reference image, we set homogeneous mechanical density.
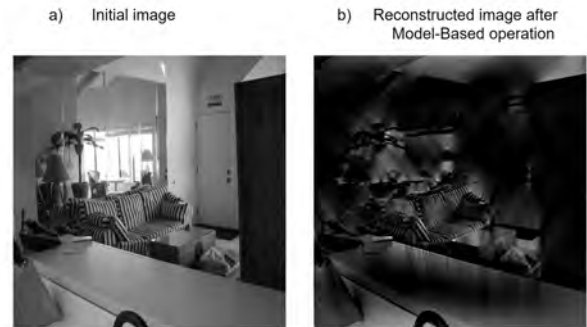
a) Initial image   b) Reconstructed image after Model-Based operation

Figure 8: Example of image from DeepMB dataset

The simulation medium was set to 2D grid of 864px by 864px with resolution of 100^{-6}m. The Field Of View (FOV), which represents the the later reconstructed image, was centred inside this k-Wave grid with the pixel grid of 416 by 416. That can be also translated to 4.16cm by 4.16 cm. The simulation grid could be seen in Fig. 9a and Fig. 9d.

Thus, the physical pressure value of minimum intensity in the image for the pressure field was set to 0 and the maximum intensity pixels was set to 10^7 a.u. The speed of sound was arbitrary picked between 1480 and
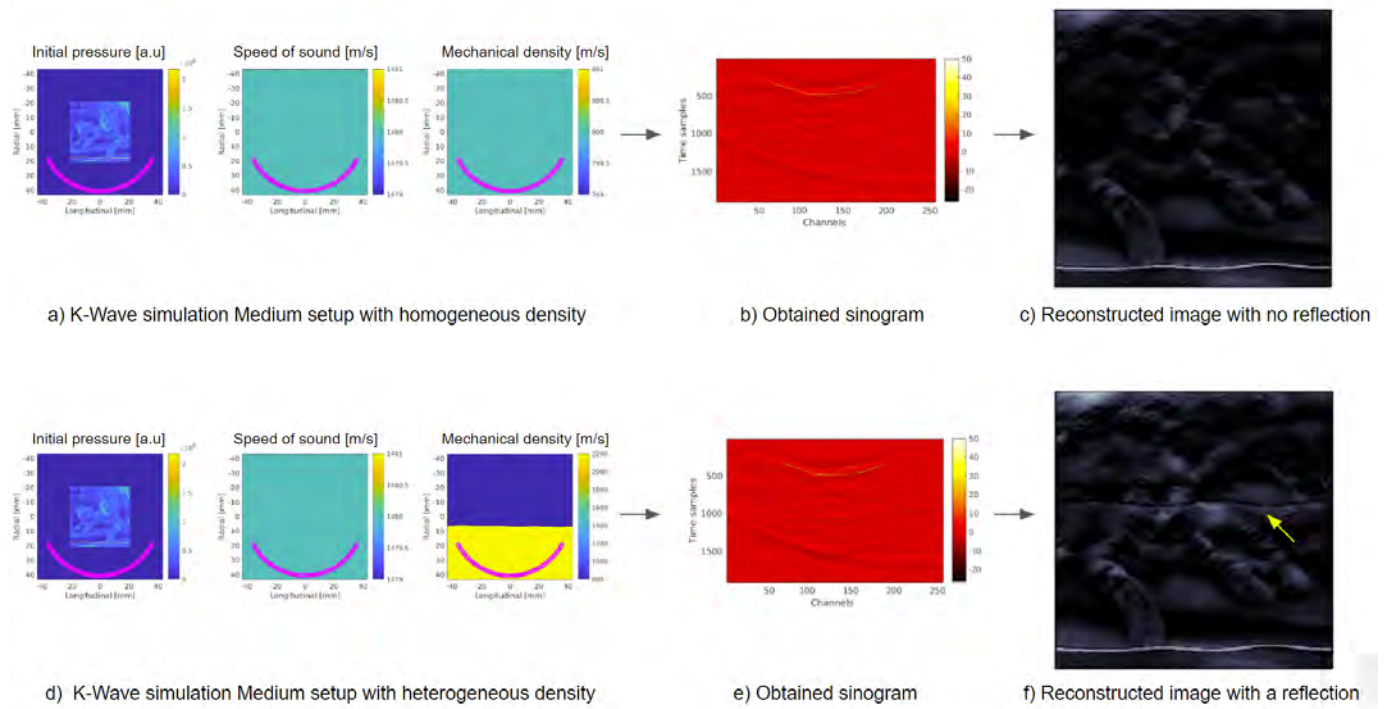
Figure 9: Synthetic data generation pipeline. (a) Simulation setup with homogeneous (uniform) mechanical density. Pink arc portrays illustrates the transducer (b) Resultant sinogram. (c) Reconstructed OA image that does not have any artifacts. (d - f) A pipeline with modelled mechanical density that results a sinogram that does have a skin reflection. Yellow arrow points to a reflection artifact
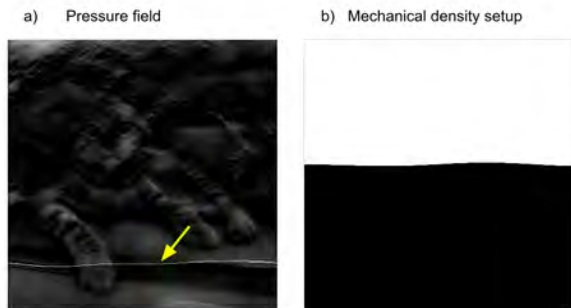

Figure 11: Example of modelled pressure field and Mechanical density for k-Wave simulation. Yellow arrow points the inserted high intensity skin imitation, similarly to in-vivo images

$1580 \, \mathrm{m \, s^{-1}}$ for whole simulation medium. For the heterogeneous mechanical density, lower value was randomly chosen between 800 and $1500 \, \mathrm{kg \, m^{-3}}$ and higher value was randomly set between 1500 and $2200 \, \mathrm{kg \, m^{-3}}$. For the homogeneous case, a uniform density was set to the value between 800 and 2200 $2200 \, \mathrm{kg \, m^{-3}}$.

Another step in simulation development is correct transducer setup. For the algorithm to successfully perform on a signal obtained from an OA machine, we had to transfer the machine parameters of iThera Acuity to k-Wave software. Thus, 256 transducer elements of size and width with accurate position were digitally copied

to the simulation with the frequency sampling rate of 40×10^6 Hz. Moreover, electrical impulse response (Chowdhury et al., 2020) that is specific to the machine was applied.

Reconstruction of the sinograms in the figures is done using DeepMB algorithm (Dehner et al., 2022).

Hence, 4000 sinogram pairs were generated using the simulation. To generate each sinogram it took 2 seconds using NVIDIA GeForce RTX 3090.

3.4. Deep learning model development

The task in this section is to map a simulated sinogram that contain reflection artifact to the simulated sinogram that does not contain reflection (Fig. 10). Hence, remove the artifact.

For this, we used a Unet with Resnet34 encoder. Segmentaion library "segmentation_models" for PyTorch (Yakubovskiy, 2020) was utilized. Bias term was included to the network; kernel size was set to 5×5 with 2×2 padding. Imagenet-based weights were applied for the incoder. Decoder depth was set to 3 layers and number of filters on the first one was 128. No activation function was used. Synthetic signal images were ranged between 0 and 1 by division of 50.

For the optimizer, Adam was chosen with Learning rate of 0.01 and LR scheduler with step size of 1, as well as gamma of 0.99. For the loss we chose Mean Squared Error.
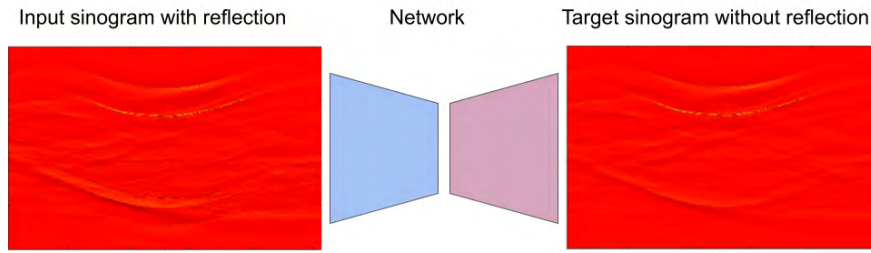
Figure 10: Mapping of input sinogram with an artifact to the artifact-free target

The data was divided into train-validation-test by ratio of 0.6-0.2-0.2. The best model for the test set was chosen by best performing epoch on validation test loss.

4. Results

4.1. Reflection artifact detection using annotated data

After training a 6 different models and testing them on separate test participant, here are the qualitative and quantitative results. All the prediction results were thresholded to 0.5, and results were converted to binary values.

4.1.1. Qualitative evaluation

Fig. 12 is the example sample of the qualitative performance of the model. The green overlay represent the reference mask of the annotator and the yellow overlay stands for the prediction.
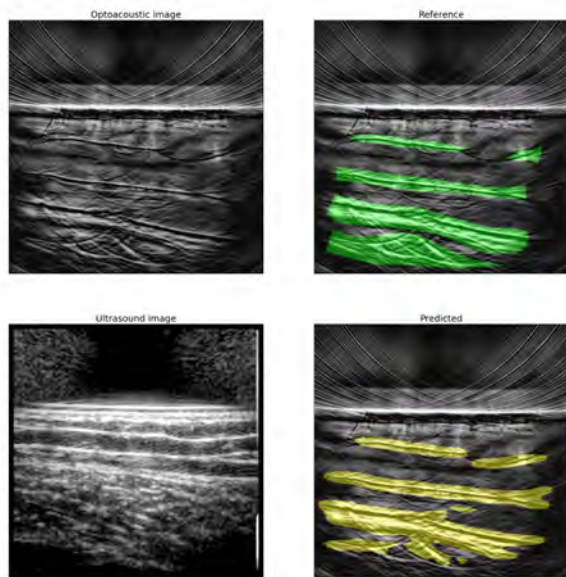

Figure 12: Qualitative performance of the network. The top and bottom left images represent the OA and US images. Right images show the reference and prediction binary masks overlaid on top of corresponding OA image. The US image was given to validate a presence of reflection artifacts.

Fig. 13 demonstrates the difference in predictions for low and high wavelength of OA images. As it was discussed in section 3.1.2, the appearance of skin reflection artifacts diminishes with increase of wavelength.

Figures 14 - 15 demonstrate network over-performing labels. The specialist confirmed that the model segments artifacts that were misannotated.
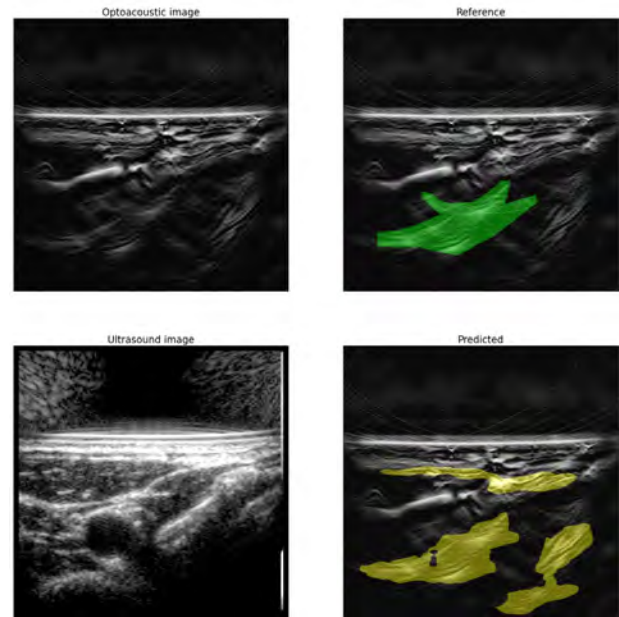

Figure 14: Network indicating reflections that were missanotated by specialist on thyroid

The network rarely made a false positive predictions. The example of such case could be noticed in Fig. 16. The misslabeling occurred only on two wavelengths of the anatomical region out of 29. Fig. 17 shows the example of the network under-performing the annotator.

4.1.2. Quantitative evaluation

In this section quantitative results are presented. In Table 2, the results for all the test participants from 6 trained models. The dice score is approximately 3 times than the random generated mask. The recall and ac-
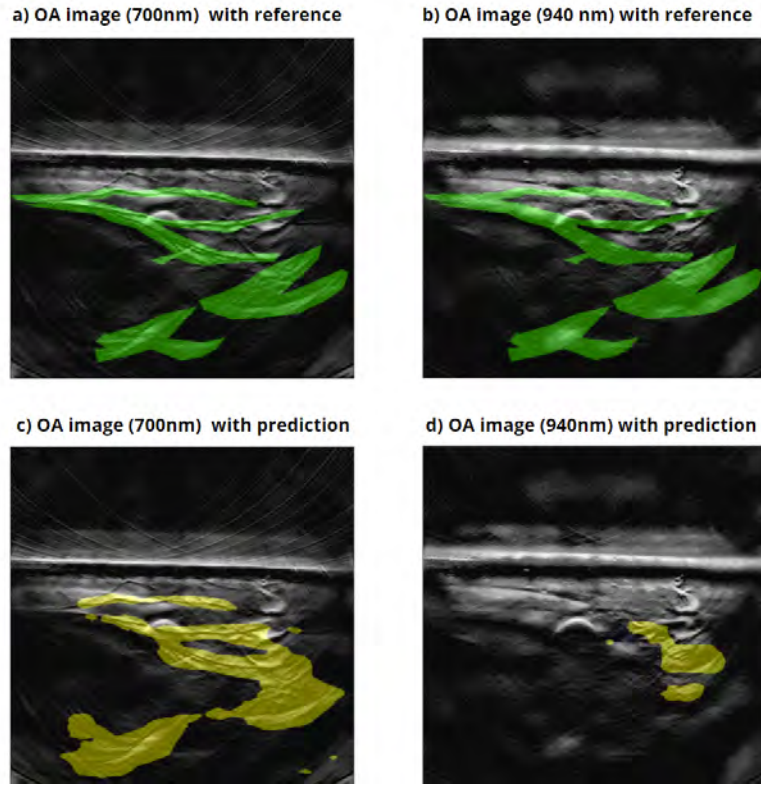
Figure 13: The comparison of an OA image predictions for 700nm and 940m. Figures a-b display low and high wavelength OA images with the same overlaid reference mask, whereas c-d shows correspond to network prediction

curacy values are significantly larger than than the random, while precision is on par.

Figures 22 - 21 are showing scores per each wavelength across all the test participants. The dice and re-call scores gradually go down while having slight increase after 900nm. The accuracy and and recall are approximately staying stable.
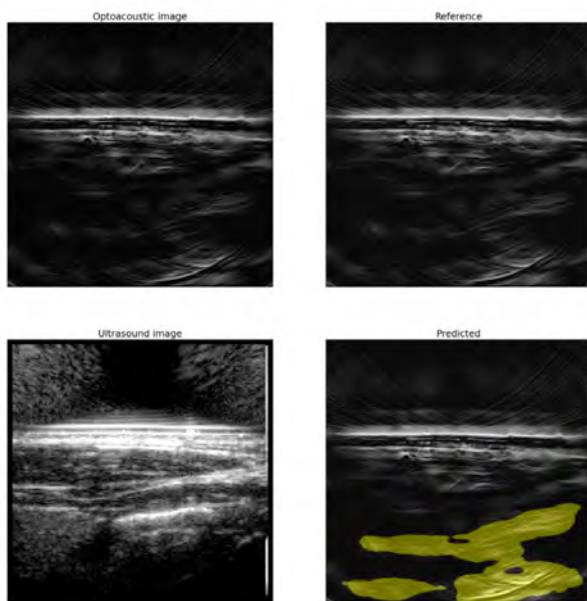

Figure 15: Example of network over-performing the annotator

		Network	Random
Specificity	**mean**	0.949	0.907
	std	0.046	0.068
Recall	**mean**	0.526	0.093
	std	0.308	0.068
Precision	**mean**	0.446	0.441
	std	0.276	0.162
Accuracy	**mean**	0.918	0.500
	std	0.044	0.001
Dice score	**mean**	0.434	0.147
	std	0.247	0.093

Table 2: Overall results for segmentation. Results for randomly generated masks were also shown for the comparison

Table 3 demonstrates the scores for each test participant. We can notice that that mean dice score is actually does not show the score more than 0.5 even among each participant. The best performing participant is actually p01, which was trained on four others and validated on the p06. The worst performing was p05 in terms of dice and recall scores. Meantime, it should be noted that amount of samples is the biggest among the six participants, hence it has less data to be trained on.

| Participant	Specificity		Recall		Precision		Accuracy		Dice score	
	mean	std	mean	std	mean	std	mean	std	mean	std
p01	0.952	0.034	0.551	0.223	0.533	0.265	0.919	0.030	0.500	0.207
p02	0.936	0.039	0.510	0.311	0.461	0.294	0.898	0.044	0.459	0.283
p03	0.908	0.059	0.724	0.287	0.366	0.206	0.901	0.053	0.459	0.218
p04	0.961	0.031	0.589	0.312	0.375	0.274	0.947	0.032	0.418	0.259
p05	0.973	0.029	0.330	0.254	0.490	0.323	0.917	0.041	0.364	0.258
p06	0.963	0.036	0.458	0.253	0.505	0.307	0.922	0.037	0.449	0.242

Table 3: Results per participant
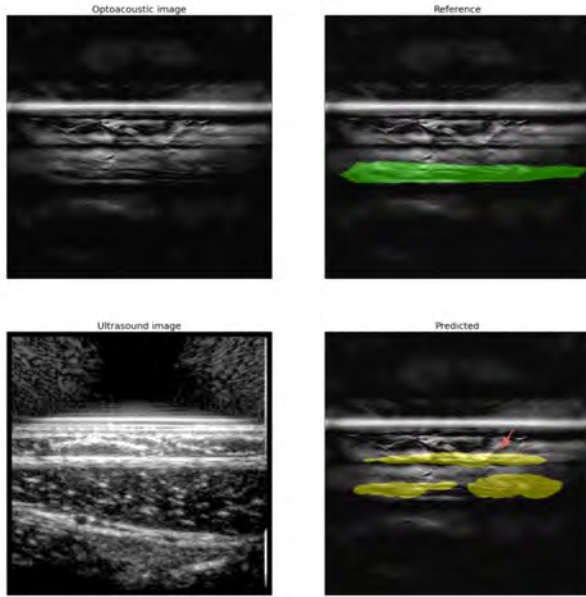
Figure 16: Case of false positive segmentation. Pink arrow represents a misslabelling by network. The impedance mismatch was falsely predicted only on two OA images with wavelengths 910 and 920

4.2. Synthetic data-based artifact removal

The network mapping on a synthetic sinogram data was performed w...

work performance on the real in-vivo images was unacceptably poor. In Fig. 23, it could be noted that after inferring a test sample sinogram, the reconstruction of the predicted image is closely resembles the target. And a reflection line was eradicated completely.

However, when the same model is inferred on the real in-vivo image, the result is mostly focused on skin itself and does not significantly affect any other structure. The same trend was noted for other samples as well.

5. Discussion

In this paper, two methods for detection and removal of the reflection artifacts were suggested. The detection method based on reflection segmentation achieved good qualitative results, while the second method based on simulated sinograms did not work on the in-vivo images.

5.1. Reflection artifact detection using annotated data

We decided to proceed with segmentation task and not with object detection, since due to the high number and incredibly complex shapes of the artifacts, the predicted bounding box could have bound half of the
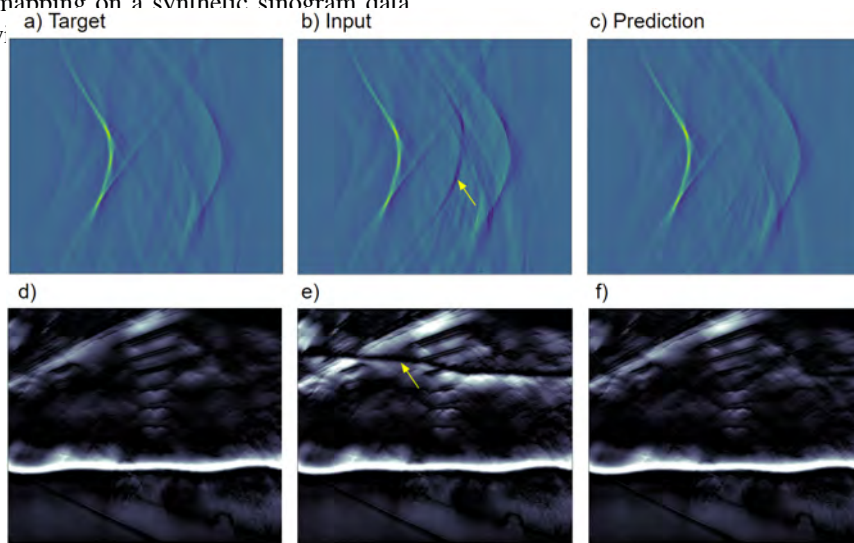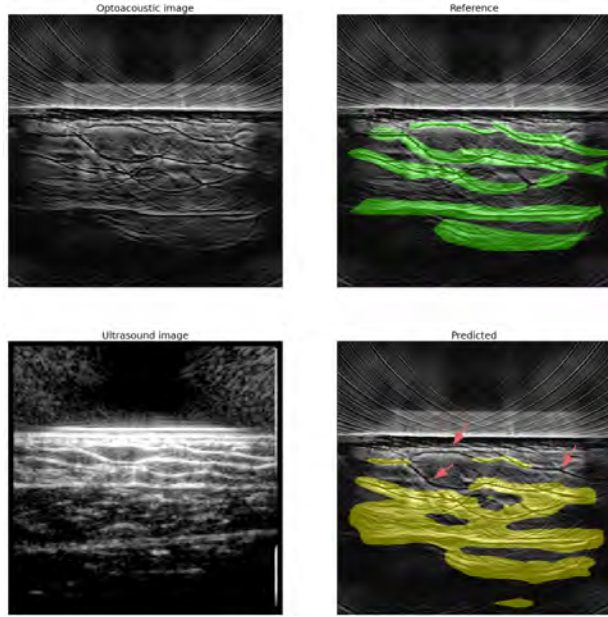

Figure 23: Network's performance on synthetic sinogram. (a-c) Target, input, prediction sinograms, (d-f) respective image reconstructions . Yellow arrows point to reflection artifact

Figure 17: False negative case. Pink arrows point to reflection areas that were not predicted correctly

Figure 18: Specificity per wavelength. The y-axis of the plot is scaled between 0.8 and 1

image. For our purpose, outline of the reflections plays a huge role, since it is important to identify whether the artifact intersects the Region-of-interest (artery, tumor, etc.) or not. That is the reason polygon labels were converted to masks. Hence, there are some limitation to this method.

The network that was trained on manually labeled artifacts gave satisfying results. In Fig. 12, it could be noticed that prediction was able to capture the reflection structure and enclose them into the segmented area. The boundaries, however, are varying in width and length
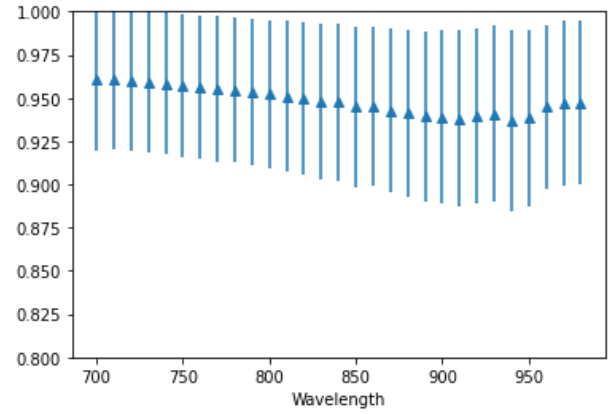
due to the fact that the exact outline of skin reflections is not given.

The other limitation is the difference in appearance between reflections on OA images obtained from low-wavelength laser pulse and high. Thus, since the reflection artifact is more obvious on the low wavelength, due to the absorption spectrum of melanin in skin, it was decided to annotate only on low wavelength, despite the fact that for high wavelengths reflection are less pronounced or absent at all. The same reference for all 29 wavelength images for the same anatomical region may complicate the training procedure, as well as resulted inference scores. In Fig. 13, it is clearly seen that performance for both high and low wavelength varies significantly, even though labels are the same.

The other interesting feature of the network is its ability to over-perform the annotator. In Fig. 14, it cold be noticed that the number of prediction areas are more
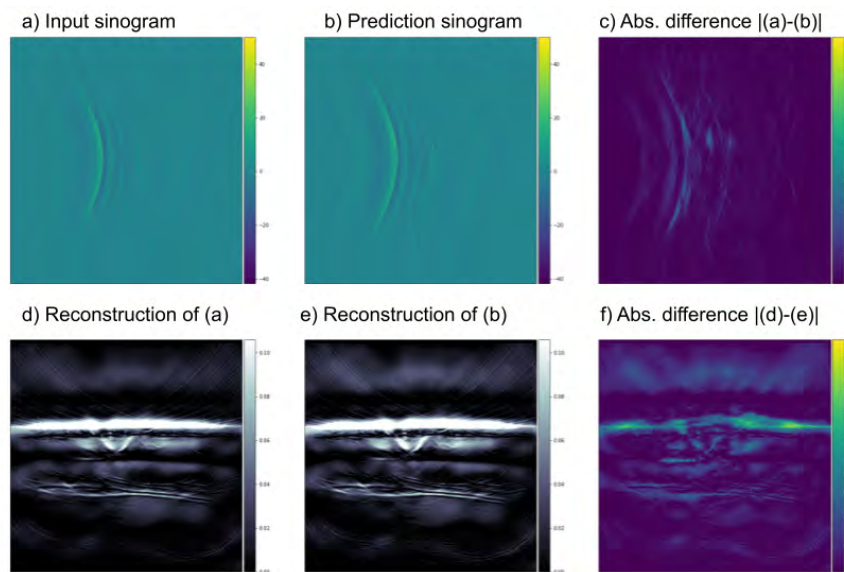

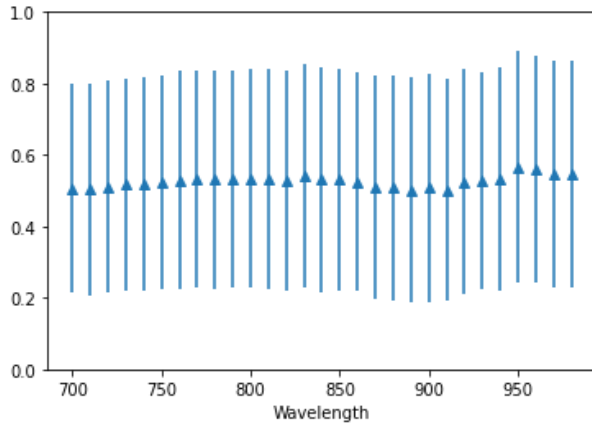Figure 24: Network's performance on in-vivo sinogram
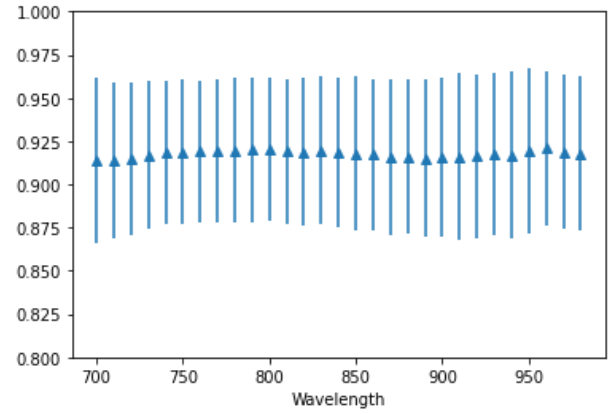
Figure 19: Recall per wavelength

Figure 21: Accuracy per wavelength. The y-axis of the plot is scaled between 0.8 and 1
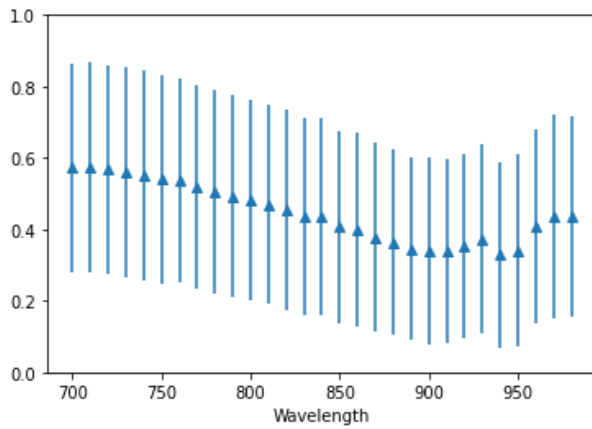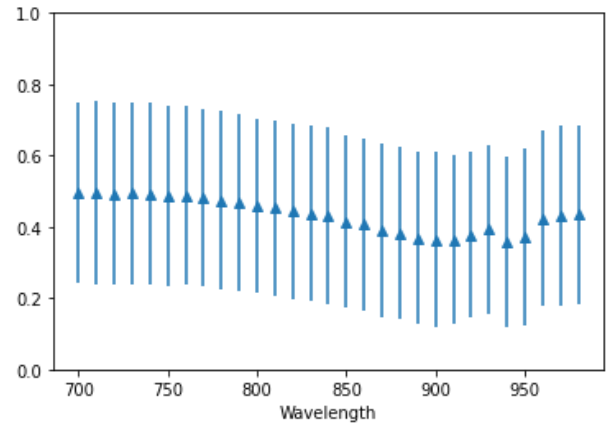

Figure 20: Precision per wavelength

Figure 22: Dice score per wavelength

than in the reference made by the OA specialist. However, those regions are not false positive cases. If you consider the ultrasound image in the bottom left, it could confirmed that those predictions are indeed the reflection artifact, meaning that the network was able to detect reflections that were missed by the annotator. The same example is showed in Fig. 15, where no reference labels were given.

After result analysis, we realized that our task closely relates to the field of machine learning called weakly supervised learning, which recently gained some more research traction. Supervised deep learning proved that it could give high accuracy results (Krizhevsky et al., 2012), considering large amount of data that was labeled correctly. However, it is not always possible to gather a high quality data and especially ground truth annotations. There could be different factors: competence of an annotator, limitation in annotation tool or inability to make a correct label due to the data complexity as it was for our annotations. In his paper, Zhou (2017) gave three types of weak supervision. One of the types is inexact supervision, where only coarse-gained labels are provided. As an example, we could consider an ob-

ject detection task, where only image-level labels were given and not the bounding boxes. Another type is the inaccurate supervision, where given annotation do not always resemble the ground truth. And the last type is the incomplete supervision. Incomplete supervision is defined by the data set that is to some degree remained unlabelled. This is exactly the case in this work, where notable amount of reflection artifacts were not annotated due to its feature intricacy in shape and appearance on different wavelengths. Miller and Uyar (1996) argues that in such cases unlabeled data could actually improve results of data-orientated algorithms.

Moreover, after going through the test samples, we noticed that the network rarely provides a false positive cases. The example of the false positive case is displayed in Fig. 16. The same time, false negative cases appear more often.

Thus, combination of border approximations during annotation, as well as wavelengths appearance difference lead to low scores during the test. Table 2 demonstrates the overall dice score, recall, precision and accuracy for all 6 test participant combined. The random generated mask scores were given for comparison. It

was found that there is a relationship between number of training data fed to the network and resulted test scores. Fig. 25 shows this relation.

It is also noted that overall dice score for OA images is increasing after 900 nm. This is phenomena occurs due to rising absorption of lipids and water at this level in the epidermis. Hence, reflection artifacts look more pronounced.
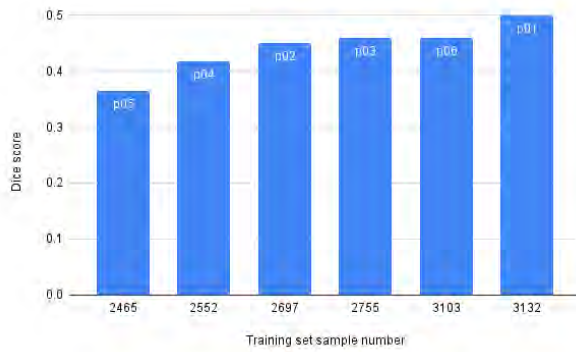

Figure 25: Number of training samples to test dice score for each participant

5.2. Synthetic data-based artifact removal

The main obstacle in deep learning is often accessibility to a data. However, recent researches proved that synthetic data based neural network training, especially in medical imaging, could elevate the results on the real samples. (Chen et al., 2021), (Yi et al., 2019), (Yang et al., 2022). Synthetic data is more accessible, whereas medical data is highly sensitive and poorly labeled. Therefore, generated data is often pre-annotated and, comparatively, cheap. This was the reason, we decide to approach our problem using the simulation approach, because there are limited amount of open source datasets in optoacoustic imaging.

It is helpful to mix the synthetic and real data during the training process. This method could also lead to significant accuracy increase (de Melo et al., 2020). However, for some types of tasks, there are could be no labels. Thus, Shi et al. (2022) fused synthetically designed needles with real photoacoustic in-vivo images of tissues, obtaining semi-synthetic dataset. After, they mapped combined images to the simulated image of needle; hence, segmenting it. In our case we also generating labels, but without fusing the real and synthetic data.

The detection of the artifact on manually annotated data showed promising results, while the synthetic based artifact approach did not work as good. Overall, the trained network showed almost perfect results on synthetic data. The modelled skin reflection was eradicated closely to flawless. However, when it came to the inference of in-vivo images from the real human participants, the network failed completely. In Fig. 24, as it

was mentioned, on the prediction of the invivo sample, there were no significant structural changes, except for some intensity changes.

The explanation to these results could be failure in domain adaptation. Domain shift is significant limitation of the current state of neural networks (Sankaranarayanan et al., 2018). It defined by inability of the models to perform on the data sets that are different from training set. Therefore, if unseen data have even slight difference in distribution, deep learning algorithm fails to generalize. There are could be several reasons for it; according to Kouw and Loog (2021), they are bias in sampling, changes in color and intensities, view angle, different distribution of noise. For our case, we speculate that there could be a anatomical distinction in skin artifact representation.

6. Conclusions

Optoacoustic imaging is a promising tool for functional imaging. Due to different optical absorption of substances, it is possible to differentiate them and quantify. However, if the region of interest, such as artery or tumor, is overlaid by reflection artifact, it could lead to misinterpretation of numerical results and could be detrimental for right diagnosis. Therefore, it is necessary to detect and remove them.

To this day, there are only three works that are focused on identifying and removing reflection artifacts using deep learning algorithms. In this paper, we proposed two new approaches for this task. One is a deep learning framework trained on manually annotated artifacts and another removal framework completely based on simulated data. Even though, the latter did not perform at all on real in-vivo samples, the detection network gives promising qualitative results.

From the segmentation method, it was found out that the network, which was weakly labelled, might overperform the annotator in detecting reflection artifacts; however, in future studies, there should be multiple annotators to prove this claim. Moreover, there is a correlation between amount of data that was used for training and test results. One of the limitations could be considered the reported metric, it does not capture the complexity of the data. It was discussed that due to reflection appearance through different wavelengths and varying segmentation boundaries, mentioned scores are not representative enough.

There are obvious uses for this technology. Firstly, it is reflection artifact detection tool that is installed into machine and gives warning when region of interest is interesting with detected reflection. Secondly, reflection identification training tool could be developed for optoacoustic application specialists. Where they could apply the algorithm to different data to learn features of the artifacts on given images.

From the synthetic removal method, we found that the network is able to remove reflection on the simulated dataset close to flawless. In the future work, novel domain shift adaptation techniques should be used.

Acknowledgments

References

Agrawal, S., Suresh, T., Garikipati, A., Dangi, A., Kothapalli, S.R., 2021. Modeling combined ultrasound and photoacoustic imaging: Simulations aiding device development and artificial intelligence. Photoacoustics 24, 100304.

Allman, D., Reiter, A., Bell, M.A.L., 2018. Photoacoustic source detection and reflection artifact removal enabled by deep learning. IEEE transactions on medical imaging 37, 1464–1477.

Beard, P., 2011. Biomedical photoacoustic imaging. Interface Focus 1, 602–631. doi:10.1098/rsfs.2011.0028.

Bell, A.G., 1880. On the production and reproduction of sound by light. URL: https://doi.org/10.2475/ajs.s3-20.118.305, doi:10.2475/ajs.s3-20.118.305.

Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A., 2020. Albumentations: Fast and flexible image augmentations. Information 11. URL: https://www.mdpi.com/2078-2489/11/2/125, doi:10.3390/info11020125.

Chen, R.J., Lu, M.Y., Chen, T.Y., Williamson, D.F., Mahmood, F., 2021. Synthetic data in machine learning for medicine and healthcare. Nature Biomedical Engineering 5, 493–497. doi:10.1038/s41551-021-00751-8.

Chowdhury, K.B., Prakash, J., Karlas, A., Justel, D., Ntziachristos, V., 2020. A synthetic total impulse response characterization method for correction of hand-held optoacoustic images. IEEE Transactions on Medical Imaging 39, 3218–3230. doi:10.1109/tmi.2020.2989236.

Cox, B.T., Laufer, J.G., Beard, P.C., 2009. The challenges for quantitative photoacoustic imaging. SPIE Proceedings doi:10.1117/12.806788.

Dehner, C., Zahnd, G., Ntziachristos, V., Jüstel, D., 2022. DeepMB: Deep neural network for real-time model-based optoacoustic image reconstruction with adjustable speed of sound. In preparation.

Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A., 2012. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

Jaeger, M., Siegenthaler, L., Kitz, M., Frenz, M., 2009. Reduction of background in optoacoustic image sequences obtained under tissue deformation. Journal of Biomedical Optics 14, 054011. doi:10.1117/1.3227038.

Kouw, W.M., Loog, M., 2021. A review of domain adaptation without target labels. IEEE Transactions on Pattern Analysis and Machine Intelligence 43, 766–785. doi:10.1109/tpami.2019.2945942.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: Pereira, F., Burges, C., Bottou, L., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

de Melo, C.M., Rothrock, B., Gurram, P., Ulutan, O., Manjunath, B., 2020. Vision-based gesture recognition in human-robot teams using synthetic data, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 10278–10284. doi:10.1109/IROS45743.2020.9340728.

Miller, D.J., Uyar, H.S., 1996. A mixture of experts classifier with learning based on both labelled and unlabelled data, in: Proceedings of the 9th International Conference on Neural Information Processing Systems, MIT Press, Cambridge, MA, USA. p. 571–577.

Nguyen, H.N., Steenbergen, W., 2020. Feasibility of identifying reflection artifacts in photoacoustic imaging using two-wavelength excitation. Biomedical Optics Express 11, 5745. doi:10.1364/boe.401375.

Nguyen, H.N.Y., Hussain, A., Steenbergen, W., 2018. Reflection artifact identification in photoacoustic imaging using multi-wavelength excitation. Biomed. Opt. Express 9, 4613–4630. URL: http://opg.optica.org/boe/abstract.cfm?URI=boe-9-10-4613, doi:10.1364/BOE.9.004613.

Ntziachristos, V., Razansky, D., 2010. Molecular imaging by means of multispectral optoacoustic tomography (MSOT). Chemical Reviews 110, 2783–2794. doi:10.1021/cr9002566.

Petrosyan, T., Theodorou, M., Bamber, J., Frenz, M., Jaeger, M., 2018. Rapid scanning wide-field clutter elimination in epi-optoacoustic imaging using comb Lovit. Photoacoustics 10, 20–30. doi:10.1016/j.pacs.2018.02.001.

Rosenthal, A., Ntziachristos, V., Razansky, D., 2014. Acoustic inversion in optoacoustic tomography: A review. Current Medical Imaging Reviews 9, 318–336. doi:10.2174/15734056113096660006.

Sankaranarayanan, S., Balaji, Y., Jain, A., Lim, S.N., Chellappa, R., 2018. Learning from synthetic data: Addressing domain shift for semantic segmentation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition doi:10.1109/cvpr.2018.00395.

Shan, H., Wang, G., Yang, Y., 2019. Accelerated correction of reflection artifacts by deep neural networks in photo-acoustic tomography. Applied Sciences 9, 2615.

Shi, M., Zhao, T., West, S.J., Desjardins, A.E., Vercauteren, T., Xia, W., 2022. Improving needle visibility in LED-based photoacoustic imaging using deep learning with semi-synthetic datasets. Photoacoustics 26, 100351. URL: https://www.sciencedirect.com/science/article/pii/S2213597922000209, doi:https://doi.org/10.1016/j.pacs.2022.100351.

Singh, M.K., Steenbergen, W., 2015. Photoacoustic-guided focused ultrasound imaging (PAFUSion) for reducing reflection artifacts in photoacoustic imaging. Opto-Acoustic Methods and Applications in Biophotonics II doi:10.1364/ecbo.2015.95390q.

Treeby, B.E., Cox, B.T., 2010. K-wave: Matlab toolbox for the simulation and reconstruction of photoacoustic wave fields. Journal of Biomedical Optics 15, 021314. doi:10.1117/1.3360308.

Yakubovskiy, P., 2020. Segmentation models Pytorch. https://github.com/qubvel/segmentation_models.pytorch.

Yang, Q., Lin, Y., Wang, J., Bao, J., Wang, X., Ma, L., Zhou, Z., Yang, Q., Cai, S., He, H., Cai, C., Dong, J., Cheng, J., Chen, Z., Zhong, J., 2022. Model-based synthetic data-driven learning (MOST-DL): Application in Single-shot T2 Mapping with Severe Head Motion Using Overlapping-echo Acquisition. IEEE Transactions on Medical Imaging , 1–1doi:10.1109/TMI.2022.3179981.

Yi, X., Adams, S., Babyn, P., Elnajmi, A., 2019. Automatic catheter and tube detection in pediatric X-ray images using a scale-recurrent network and synthetic data. Journal of Digital Imaging 33, 181–190. doi:10.1007/s10278-019-00201-7.

Zhou, Z.H., 2017. A brief introduction to weakly supervised learning. National Science Review 5, 44–53. doi:10.1093/nsr/nwx106.

Federated Learning for Multimodal Brain Tumour Segmentation

Ebaneo E. Valdez Kao, Salman Mohammadi

Canon Medical Research Europe LTD, 2 Anderson Pl, Edinburgh EH6 5NP, United Kingdom

Abstract

Deep learning (DL) is having an increased impact in different areas of sciences and daily life. Creating a state-of-the-art Deep Learning model requires collaboration from multiple actors that contribute with high-quality data. However, some technical, ownership, and data-privacy-related obstacles slow down the free collaboration of datasets.
Federated Learning (FL) was devised to allow decentralized collaboration in the Deep Learning community without the need for any data samples to leave its institution. In this work, we explore how multimodality impacts Federated Learning semantic segmentation by creating a new algorithm that allows training an FL system with any number of modalities, independently from the DL architecture chosen.

Keywords: MAIA master, Federated Learning, Multimodality, Segmentation, multimodal FedAvg, Federated Tumour Segmentation Challenge (FeTs)
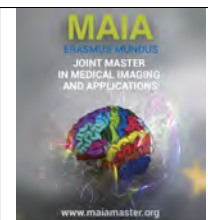
Feature registration algorithms for the correlative study of bone mineralized fibrils with small-angle scattering tensor tomography and ptychographic X-ray computed tomography

Alexandru-Petru Vasile[a], Mariana Verezhak[a,b]

[a]*Photon science division, X-Ray tomography group, Paul Scherrer Institute, Villigen PSI, Switzerland*
[b]*Department for electrical engineering and information technology, X-Ray tomography group, ETH Zürich, Zürich, Switzerland*

Abstract

Recent advancements in synchrotron X-ray tomography allow for the imaging of structures of the human anatomy that eluded the medical world in the past. By harnessing the brilliance and coherency of the Swiss Light Source, scientists have been able to retrieve information about the organization of collagen fibrils and mineral nanocrystals that make up the innermost organizational units in the hierarchical structure of bone. Small-angle scattering tensor tomography and ptychographic X-ray computed tomography were used to acquire both large field of view and high resolution data from the same bone sample volume. In this work, several computational methods are proposed for solving the problem of registration of the two tomographic volumes that contain real and reciprocal space information, respectively. Such study provides comprehensive understanding of the mineralized collagen fibrils orientation in the multi-scale organisation of bone.

Keywords: Nanoimaging, X-Ray ptychography, Small-angle scattering tensor tomography, Orientation, Correlation, Registration, Deep learning, Fourier analysis

1. Introduction

Bone is a hard biological tissue that constitutes part of the skeleton of the vast majority of vertebrates. The inner structure of the various types of bone is the result of evolution as well as adaptive processes that take place during an individual's lifetime. The study of bone is crucial from a medical point of view, understanding the structural characteristics of bone as well as the different phases present in this very complex biological material at different scales can shed new light in the development of various tools for diagnostic and treatment of numerous pathologies (Verezhak (2016)).

Furthermore, information derived from such scientific inquiries may also serve disciplines other than medical sciences, for example archaeology which concerns itself with bone analysis in several of its sub-disciplines. Material science and engineering often draw inspiration from the organization of biological structures and even architecture benefits from new findings in the study of bone as could be seen in the works of one of the greatest architects in European history, Antoni Gaudí, who drew heavy inspiration from the skeletal structure of various organisms in order to achieve some of the most impressive feats of architecture ever designed.

1.1. Hierarchical structure of bone

At the *macro-level*, bone is seen as an organ (Figure 1 - I), an integral part of the human body with the morphology of long, short, sesamoid, irregular or flat. The size spectrum of human bones starts at the millimeter level with small bones such as the the ones in the middle ear and goes up in the hundreds of centimeters with the longest bone being the femur. At birth, the bone count of a human fetus sits at 270 and, at full development, the human adult boasts 206 bones (Verezhak, 2016). In order to study bones at this scale, a number of modalities

---

*Please address correspondence to Mariana Verezhak or Alexandru-Petru Vasile
Email addresses: `alexandru.vasile@psi.ch` (Alexandru-Petru Vasile), `mariana.verezhak@psi.ch` (Mariana Verezhak)
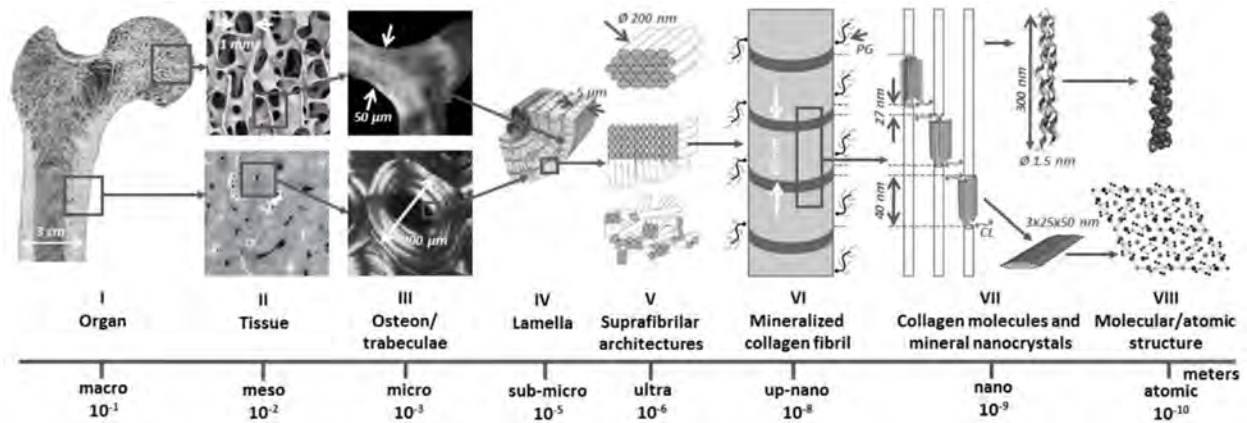
Figure 1: Bone hierarchy (example of human femur): I - photo of longitudinally cut femur, II - SEM of trabecular and cortical tissue, III - polarized light optical microscopy of a single trabeculae and osteon, IV - schematic representation of lamella, V - types of suprafibrular organizations (Weiner and Wagner, 1998), VI - collagen fibril with 67 nm periodicity (pg - proteoglycan molecules), VII - collagen molecules and mineral nanocrystals with gap (40 nm) and overlap zones (27 nm), CL - cross-links, VIII - molecular and atomic structure of the principal components (Verezhak, 2016).

are routinely used such as absorption X-ray imaging or light microscopy (LM) (Georgiadis et al., 2016).

At the next hierarchical level of bone (Figure 1 - II), the cortical or trabecular nature of bone can be seen. Cortical bone provides strength to the structures in which it is incorporated as it is a very dense type of osseous tissue. This is why it can be found in the shell of long bones and in the extremities of flat ones, providing these with stiffness and robustness. This being said, in this work the data used comes from a trabecular bone sample. By contrast, trabecular bone is more porous, hence lighter than its cortical counterpart, providing flexibility and reinforcement to the structures it is a part of, such as the inner part of flat bones. Trabecular bone also often accommodates marrow, where the process of hematopoiesis (the production of blood cells) takes place (Verezhak, 2016). For the study of mesostructure, the following imaging modalities are being used: LM, X-ray imaging and scanning electron microscopy (SEM) (Georgiadis et al., 2016).

Further down the structural scale, at the *microstructure level* (Figure 1 - III), the structural units of cortical and trabecular bone can be observed. In the case of cortical structures, these are called *osteons* and for trabecular bone, *trabeculae*. Osteons have a cylindrical shape and accommodate blood vessels and nerve fibers inside a canal that runs along their central region. Canals from different osteons are connected to each other and to marrow and are surrounded by lamellae. The *lacunae* are cavities in which mature osteocytes (bone cells) reside and they are connected by *canaliculi*, channels filled with fluid that provide a network for communication and nutrition. Trabeculae, on the other hand, arrange themselves primarily along the direction of the most significant mechanical stress in bone. They are making up a porous network interconnected by canaliculi similar to their cortical counterparts. Inside the pores of tra-

becular tissue (which generally measure from one to a few mm) resides bone marrow, bone cells and fat tissue. At this level, the main types of bone cells can also be differentiated: *osteocytes* (maintain osseous tissue), *osteoblasts* (constitutes the bone matrix), osteogenic cells (stem cells) and osteoclasts (cells that resorb bone) (Verezhak, 2016). Light microscopy can no longer be used to study bone at this organizational level, so SEM and X-ray modalities become suitable. Among the X-Ray modalities ptychographic X-Ray computed tomography (PXCT) can be used from this organizational level (Dierolf et al., 2010; Holler et al., 2014), this modality has a high importance for this project and will be explained to a greater extent in later chapters.

Woven and lamellar bone can be found at the *sub-microstructural level* (Figure 1 - IV). Woven bone is present during the developmental stages of osseous tissue or during the first stages of repairing process after a fracture. It is made up of unorganized collagen fibers. Lamella is a unit of osseous tissue that is created by osteoblasts after woven bone is resorbed by osteoclasts. The process of bone remodeling is crucial for the adaptive function of bone that is essential to growth and to changes that occur as it experiences different types of stress(Verezhak, 2016). The imaging techniques that are able to image features at the sub-microstructural level are the same as in the last organizational level.

At the *ultrastructural level* of the bone tissue hierarchy (Figure 1 - V) the *architectures* in which collagen fibrils organize themselves in become apparent. Four main types of organizational patterns are proposed in the work of Weiner and Wagner (1998). These patterns are called: parallel, plywood-like and radial. Apart from SEM, for studying bone at this level, coherent X-Ray diffraction imaging, transmission electron microscopy (TEM) and small angle X-ray scattering (SAXS) can be used. As at this organizational
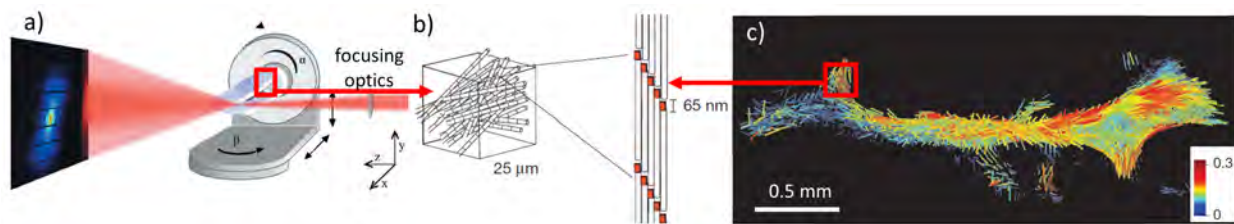
Figure 2: Small-angle scattering tensor tomography: a) SASTT experimental setup includes focused beam, scanning in xy plane, with rotations around β and tilts around α, and a characteristic bone small-angle X-ray scattering pattern (Liebi et al., 2018); b) mineralized collagen fibrils with characteristic gap and overlap zones; c) orientation of bone ultrastructure (Liebi et al., 2015).

level one can directly image the orientation of collagen fibrils, several orientation-specific techniques are commonly used: polarized light microscopy (PLM), polarization sensitive second harmonic generation imaging (pSHG), SAXS/WAXS (small/wide-angle X-ray scattering), SASTT (small-angle scattering tensor tomography) (Liebi et al., 2015), electron diffraction, etc (Georgiadis et al., 2016).

Mineralized collagen fibrils can be found at the *up-nanostructural level* (Figure 1 - VI). Individual fibrils are routinely visualized at this level by bright-field TEM, presenting a specific periodic bright-dark contrast that comes from the gap-overlap zones of the mineral population. The SAXS and X-ray crystallography (XRD) are also providing practical information about the collective average size and orientation of the fibrils (Georgiadis et al., 2016).

The *mineral nanocrystals* and *collagen molecules* can be differentiated at the level known as the *nanostructure* (Figure 1 - VII). The molecules are comprised of polypeptide chains that form a right-handed triple-helical structure and have a dimensionality of 300 nm in length and 1.5 nm in thickness. Nanocrystals apparent at this scale have sizes that vary according to the arrangement of the collagen molecules around them, the anatomical location of the tissue they are a part of, the species, age, diet and even various pathologies (Verezhak, 2016). Imaging structures at this level is very challenging and a few examples of such work can be found using automated crystal orientation mapping (ACOM) TEM (Verezhak et al., 2018), coherent diffraction imaging (CDI) (Verezhak, 2016),(Jiang et al., 2008), XRD, SAXS, pair distribution function (PDF) analysis (Verezhak, 2016).

The last organizational level considered in this framework is the *molecular and crystalline level*(Figure 1 - VIII), which contains information on the structure of the collagen and nanocrystals mentioned in higher organizational levels from a molecular and crystallographic point of view. At this level, concepts such as the degree of crystallinity, mineral crystal phases and crystal-chemistry aspects can be investigated as well as possible defects within the lattice. This can be achieved with PDF analysis, XRD (Verezhak, 2016) and HR-TEM

(high resolution transmission electron microscopy) (Kis et al., 2019). In order to obtain information about orientation at this organizational level, electron diffraction is usually the modality of choice (Georgiadis et al., 2016).

1.2. Small-angle scattering tensor tomography

A relatively new technique in the world of X-ray tomography, small-angle scattering tensor tomography (SASTT) (Liebi et al., 2015) offers the information about collective 3D orientation and organisation of samples principal components, i.e. bone fibrils, within large volumes (few hundred of μm). Each element of SASTT is no longer a voxel, but a tensor describing the average orientation of the structure of the sample probed by the beam of the given size (Guizar-Sicairos et al., 2020).

This technique was originally developed and demonstrated by Liebi et al. (2015) on a human trabecular bone sample. The diagram in 2 shows the concept behind SASTT acquisition: the incident X-ray pencil beam scans the sample to acquire a single projection from the set of scanning positions recorded as SAXS signal on the detector. The projections are collected at different tomographic angles (around the *y* axis) as well as at various tilts (around *x* axis). As illustrated, in this case the information that leads to the pattern registered on the detector comes from the nano-scale organization of the collagen fibrils that make up the bone. The features that are probed by using this method range from a few hundred nanometers to a few nanometers.

As SASTT offers insight into the local orientation of the ultrastructure of the sample, the data has to be reconstructed in such a way that the orientation is visible. One such representation is a 3D arrow (glyph). In order to achieve such a representation, Liebi et al. (2018) propose to reconstruct the data by modeling the 3D reciprocal space collected by the detector with *spherical harmonics*. These are a set of 3D basis functions used to model volumetric shapes. The parameters are the degree (*l*), the order (*m*) and the coefficients (a_0, a_1, a_2, a_3) of the spherical harmonics. The first coefficient a_0 is used to describe the symmetry of the sampled neighbourhood, while the other three encode information about its anisotropy. Following the reconstruction, the orientation is described in spherical coordinates using the
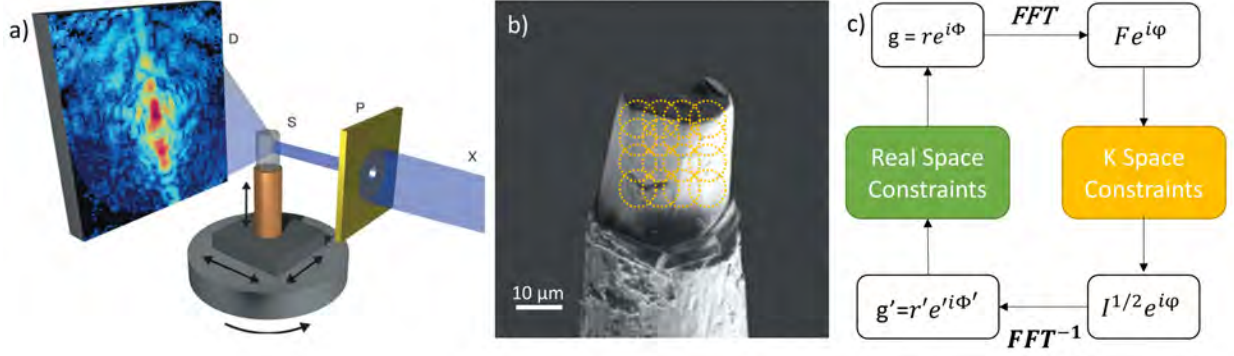
Figure 3: Ptychographic X-ray computed tomography: a) PXCT experimental setup, X – X-ray beam, P – pinhole, S – sample, D – detector. The sample is scanned in the plane perpendicular to the beam direction in the set of overlapping positions b) to acquire a single projection. Such projections are recorded for all tomographic rotation angles (Dierolf et al., 2010). c) iterative phase retrieval algorithm used for ptychographic reconstructions (Faulkner and Rodenburg, 2004).

azimuthal angle ϕ and the polar angle θ and a set of above-mentioned spherical harmonics coefficients. In order to obtain the 3D arrow representation, this information was converted to the Cartesian coordinates using the following transformation:

$$
\begin{aligned}
x &= \sin\theta\cos\phi; \\
y &= \sin\theta\sin\phi; \\
z &= \cos\theta.
\end{aligned}
\tag{1}
$$

In order to obtain the information about the local *symmetry*, the vectorial components were multiplied by the principal spherical harmonic coefficient, a_0. In addition, to encode the information about the *degree of orientation* (i.e., how strongly the collagen fibrils are oriented in a certain direction), the vector is multiplied by the ratio of the anisotropic components to the full set of components of the spherical harmonic coefficients, $\frac{(a_1+a_2+a_3)^2}{(a_0+a_1+a_2+a_3)^2}$ (Liebi et al., 2018).

1.3. Ptychographic X-Ray computed tomography

Ptychography is a lensless imaging method, which by iterative reconstruction of the coherent interference patterns scattered from the sample in the set of overlapping positions, provides the reconstructed image. Ptychography can be used with X-rays, visible light, electrons and extreme ultraviolet radiation. Unlike absorption-based imaging that is predominately used in the clinical world today, ptychography is working with diffraction (scattering) data.

As shown in Figure 3a, in ptychography, a focusing element or a pinhole is used to select coherent and pencil-shaped illumination obtaining a diffraction pattern on the detector. The scanning of the specimen at hand is performed such that the different recorded regions overlap (see Figure 3b) thus leading to redundancy in the acquired data. This is a crucial aspect that ensures for the iterative reconstruction (summarised in Figure 3c) of both the sample and the illumination function (Diaz et al., 2012).

Negative effects introduced by limited numerical aperture or artifacts produced by lenses are avoided when using ptychography as it is a lensless method. This aspect also accounts for the high resolution (below 10 nm) that can be achieved with ptychography as it is only limited by the maximal angle of scattered radiation that interacts with the detector and the radiation damage of the sample.

Another advantage of ptychography is its resistance to noise due to the fact that even though it requires the same number of counts as a conventional image, these counts are distributed over a large number of diffraction patterns because dose fractionation applies in such a case (Thibault and Guizar-Sicairos, 2012).

In order to reconstruct the data of ptychographic X-Ray computed tomography (PXCT), two mathematical objects have to be retrieved: the illumination and the object. The ptychographic data reconstruction is performed by using phase-retrieval algorithms. In this work the combination of

On of the two methods used in this work, the difference map algorithm, was developed by Thibault et al. (2009). Since this is not the scope of this work, the algorithm will not be described at length but as a summary it is comprised, as most inverse-problem algorithms, of a minimization of a cost function based on a set of constraints. In the case of ptychographic reconstruction, the constraints are imposed by the diffraction patterns (the information gathered from the detector). In order to obtain the missing phase information of the image, the algorithm starts with a random object (including the amplitude and the phase) and random illumination. Phase, the imaginary part of the complex wavefront that interacts with the sample, is then iteratively refined by series of back and forward Fourier transforms of the object, while updating the amplitudes using measured intensity from the detector. This process reconstructs the illumination function as well as the object function.

1.4. Goal of the project

The main goal of this project was to develop the feature registration algorithm that provides a correlation between the PXCT and the SASTT data, acquired on the same bone sample volume. Such feature registration required not only 3D spatial registration, but also finding and angle of rotation of one volume with respect to another around the tomographic axis. Such work would demonstrate the complementarity of the two techniques, PXCT providing high resolution and SASTT providing large field of view.

As SASTT data is represented as tensorial volume, describing the reciprocal space of the sample and PXCT is providing with real space gray scale tomography, a common set of features had to be decided upon; these features had to represent the 3D orientation of both SASTT tensors and fibers visible in ptychographic data. Hence, SASTT tensors could be correlated with information from the gray scale values of the PXCT voxels. For this, the SASTT tensors have been transformed from the spherical coordinate system to Cartesian coordinates and several methods were investigated for the extraction of orientation information from the collagen bundles visible in the ptychography data. One of the strategies for finding a correlation was to obtain a representation from autoencoder neural networks and it involved describing the two volumes in the latent space.

This work will not only benefit the fundamental understanding of bone tissue but might play a role in the development of new diagnostics and treatment tools. In addition, it might impact other fields of biophysics and material science by validating the physical meaning of a tensor in small-angle scattering tensor tomography with information from real-space data obtained from ptychography. Moreover, the algorithm proposed in this paper could come to the aid of other similar projects in which a correlation between 3D information from different imaging modalities has to be investigated. For example, one other field in which the work presented in this thesis could be of use is the one of neural circuit research. By studying the orientation of fibrilar structures one could infer useful characteristics about the electrochemical interactions between bundles of neurons, helping or bypassing the need to segment individual neural cells.

2. State of the art

At a first glance, this project seems to be an image registration work but thanks to the nature of the data, it is clear that the correlated features were never going to be images. After a thorough literature review, we concluded that most of the work done in the field of correlative imaging is concerned with registration of images obtained from different modalities but images nonetheless. The main hurdle that had to be overcome during this project was the fact that one of the volumes that needed to be correlated was not an image but a volume of tensors.

To the best of our knowledge, such a correlation between these two specific data sets is attempted for the first time in this work. The only known work Guizar-Sicairos et al. (2020) described the validation of small-angle scattering tensor tomography by using scanning SAXS, at the cSAXS beamline of the Swiss Light Source (SLS) Synchrotron. For the SAXS acquisition, the sample used for SASTT was cut into slices, since this method is not suitable for 3D analysis modality does not. Since in the present work, ptychographic X-ray computed tomography is used, the information about the bulk 3D volume is obtained in non-destructive manner, thus having the potential to further validate the 3D capabilities of this method.

In a similar manner, Khan et al. (2015) have developed a method for analysing light microscopy data for validating diffusion MRI (d-MRI). For this, stacks of confocal microscopy images of hippocampal tissue were acquired and then the orientation of the tissue is analysed and compared to data acquired via d-MRI. The features describing the orientation of the tissue were extracted in this study is through structure tensor (ST) analysis.

Rezakhaniha et al. (2012) have also used structure tensor analysis in order to study the orientation and waviness of the collagen structures in arterial adventitia from images of the confocal laser scanning microscopy. The structure tensor analysis is also mentioned by Püspöki et al. (2016) as a robust method for the analysis of tissue directionality in biological images. This algorithm was developed by Daniel Sage and his team at EPFL, Switzerland. In their implementation, the structure tensor (Equation 2) is analysed in a neighborhood around the each pixel p to extract information about the local orientation $\theta(p)$ by processing it through Equation 3.

$$S(p) = \begin{bmatrix} (I_x(p))^2 & I_x(p)I_y(p) \\ I_x(p)I_y(p) & (I_y(p))^2 \end{bmatrix} \quad (2)$$

where $I_x(p)$ and $I_y(p)$ are the local gradients of the image along the considered directions (x and y respectively).

$$\theta(p) = \frac{1}{2} \arctan 2 \frac{\langle I_x(p), I_y(p)\rangle_w}{\langle I_y(p), I_y(p)\rangle_w - \langle I_x(p), I_x(p)\rangle_w} \quad (3)$$

where $\langle I_x(p), I_y(p)\rangle_w = \iint_{\mathbb{R}} w(x,y)I_x(p)I_y(p)dxdy$ is the weighted inner product of the two gradient functions, and $w(x,y) \geq 0$ is the weighting function that delineates the region of interest.

Recently, deep learning methods for orientation analysis have also started to appear in the scientific commu-
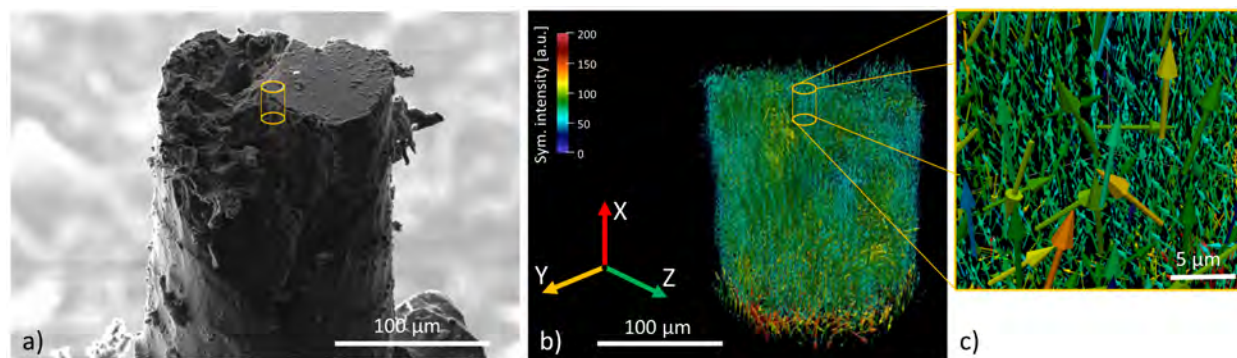
Figure 4: Trabecular bovine bone sample: a) scanning electron microscopy image of the sample used for SASTT analysis, b) orientation map of mineralized fibrils as obtained by SASTT. Yellow cylinders highlight the sample volume extracted for PXCT investigation.

nity. Schmarje et al. (2019) propose a method for segmenting the fibers in an image based on their degree of organisation. In this work, fiber bundles were classified according to how much oriented they were, the pixels of an image were potentially belonging to three classes: 1) pixels belonging to an area of fibers with similar orientation, 2) pixels belonging to an area of dissimilar orientation, and 3) pixels belonging to an area that is not of interest due to a lack of fibers in that region. The authors have compared the result of this approach to other methods that can indicate whether an area is strongly oriented or not, such as Fourier space analysis. The downside of such techniques is that they do not tackle the quantitative estimation of orientation itself.

3. Material and methods

The samples were prepared and measured at the cSAXS beamline of the SLS prior to the beginning of this thesis project.

3.1. Sample preparation

The bovine bone samples were obtained from the local butcher (Berchtold Fleisch AG, 5037 Muhen, Switzerland) with authorisation of use for the scientific purposes. The animal is female bovine, 486 days of age with the ID of TVD 120.1370.6096.7. Both cortical and trabecular samples were extracted and cut to the dimensions of $1x1x1\ mm^3$ from the tibia bone with a high precision circular diamond saw (Buehler Isomet low speed saw): the medial cortical quadrant from the mid-diaphysis for the cortical sample and the distal mid-epyphysis for the trabecular sample. The samples were fixed in ethanol 70% for 10 days, subsequently dehydrated (48 h in ethanol 100%) and slowly dried in a desiccator.

The samples for SASTT were reduced in size using the lathe system (Holler et al., 2020), resulting in the cylinder shape of ~ 160 μm in diameter and 175 μm in height (see example for trabecular sample in Figure 4a).

3.2. SASTT data acquisition

The SASTT was performed in a similar way to (Liebi et al., 2015) at X-ray photon energy of 11.2 keV with the beam focused to ~5x5 μm² (HxV). SAXS and WAXS (wide-angle X-ray scattering) signal was simultaneously acquired. The SAXS signal was recorded with Pilatus 2M detector at 2 m from the sample. The flight tube was under vacuum. The sample volumes were measured with 5 μm scanning step size, azimuthal angular spacing of 6° between 0° and 180° at seven different tilt angles of the tomographic axis between 0° and 45°. We acquired in total 345 projections per sample, by scanning 35 × 45 positions in each (including some buffer space around the sample). This results in 540,000 scattering patterns with a total measurement time of 17.6 h. Standards were used for the data calibration (glassy carbon, LaB6, AgBe). The data was radially integrated and the tensor tomograms were reconstructed, see Figure 4b,c.

In order to visualize the tensorial data that is provided by the spherical harmonics based reconstruction for SASTT, 3D rendering was used, producing results as the ones illustrated in Figure 4b,c. This was performed by ParaView software. The array of orientations was comprised of the θ and ϕ angles and the principal coefficient of the spherical harmonic, a_0, measuring the symmetry of the sampled region.

3.3. PXCT data acquisition

After SASTT data was successfully acquired, two 15 μm in diameter and 40 μm in height cylinders were extracted from the centers of each SASTT volumes using plasma focused ion beam (ScopeM, ETH) and focused ion beam (PSI). Then the ptychographic tomography data was acquired from each cortical and trabecular volumes using both flOMNI (Holler et al., 2014) and OMNY (Holler et al., 2018) setups in order to compare the effect of radiation damage on the spatial resolution in PXCT. In addition, 2 cortical and 2 trabecular fresh control samples were prepared for PXCT. In this work,
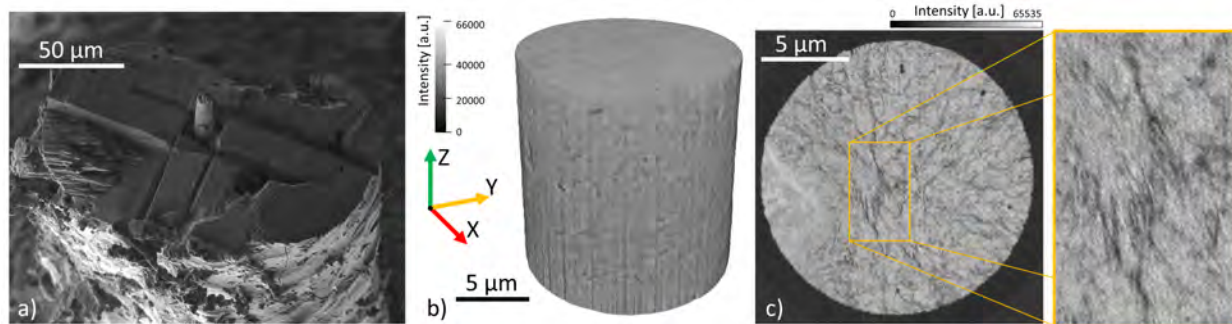
Figure 5: PXCT sample volume: a) scanning electron microscopy image of the sample extracted for PXCT analysis, b) 3D rendering of the ptychographic reconstruction, c) axial slice of the ptychographic volume. Individual collagen fibrils are visible as can be observed in the zoomed patch.

the results will be presented on one sample, the trabecular sample that was measured by SASTT and PXCT with flOMNI setup.

PXCT data was acquired with the following parameters using photon energy of 6.2 keV. We obtained $20x15\,\mu m^2$ FOV with the step size of $0.8\,\mu m$ and count time of 0.1 s that resulted in 1030 projections per sample. Data was acquired with Eiger 1.5M detector at 5.271 m downstream of the sample.

Ptychographic projections were reconstructed in an area of 700×700 pixels of the Eiger 1.5M detector, resulting in a pixel size of 20.07 nm using 300 iterations of the difference map (DM) algorithm (Thibault et al., 2009) followed by 800 iterations of a maximum likelihood (ML) refinement (Thibault and Guizar-Sicairos, 2012).

For tomography, 1030 projections equally spaced over a 180-degree angular range were recorded. The phase of the reconstructed projections was then postprocessed in terms of alignment and removal of constant and linear phase components (Guizar-Sicairos et al., 2011) (Guizar-Sicairos et al., 2015), and a modified filtered back projection algorithm was applied for the tomographic reconstruction.

The 3D spatial resolution was estimated by Fourier shell correlation (FSC) with the ½-bit threshold criterion. For this we split the projections into two independent data sets, each with double angular spacing, by taking either the even or odd angular projections. With this we then compute two independent tomograms. The FSC between these two tomograms is calculated and the point where the FSC intersects the 1/2 bit threshold defines the resolution (Heel and Schatz, 2005). The 3D spatial resolution of the trabecular volume was 33 nm (see supplementary Figure 1).

3.4. Feature registration pipelines

In order to perform the correlation between the two data sets, it has been necessary to reach a common set of features taking into account the nature of the end result of the reconstruction, namely: a gray scale value

volume for PXCT and a tensorial volume for SASTT. The first challenge is therefore to convert both data sets to a comparable representation and appropriate dimensionality. In the current work, we propose three distinct pipelines to accomplish this task, that are discussed at length below:

- 3D structure tensor analysis;

- Fourier spectrum analysis;

- Deep feature correlation.

3.4.1. 3D structure tensor analysis

Using the structure tensor for extracting the information about orientation is a technique that analyses the derivatives in the three directions of a tomographic data set. This method was chosen as a first approach to deduce the main orientations of the collagen fibril bundles within the real space gray-scale volumes from PXCT.

3.4.1.1. PXCT data post-processing. The PXCT data needed to be adapted prior to feature extraction.

Apodization. In order to remove the artifacts that were located beyond the edges of the region of interest of the PXCT volume, an apodization step was introduced. To achieve this, Otsu thresholding was chosen due to its satisfactory performance and speed. However, there were still some areas inside the region of interest that were segmented out as Otsu's algorithm is a histogram-based technique and it does not take local information into account. Hence, thresholding has to be followed by a closing binary morphology operation. In the next step, the array was modified by use of padding and cropping around the region of interest so that from a dimensionality of 743x864x864 it becomes 750x750x750 without sacrificing any information of interest. This new dimensionality allows for straightforward physical interpretations of the data: since the voxel size is approximately 20 nm in the ptychographic data set, the total PXCT volume is a cube with the dimensions of $15x15x15\,\mu m^3$. The SASTT volume has a tensorial voxel size of 5 μm so this means that the PXCT
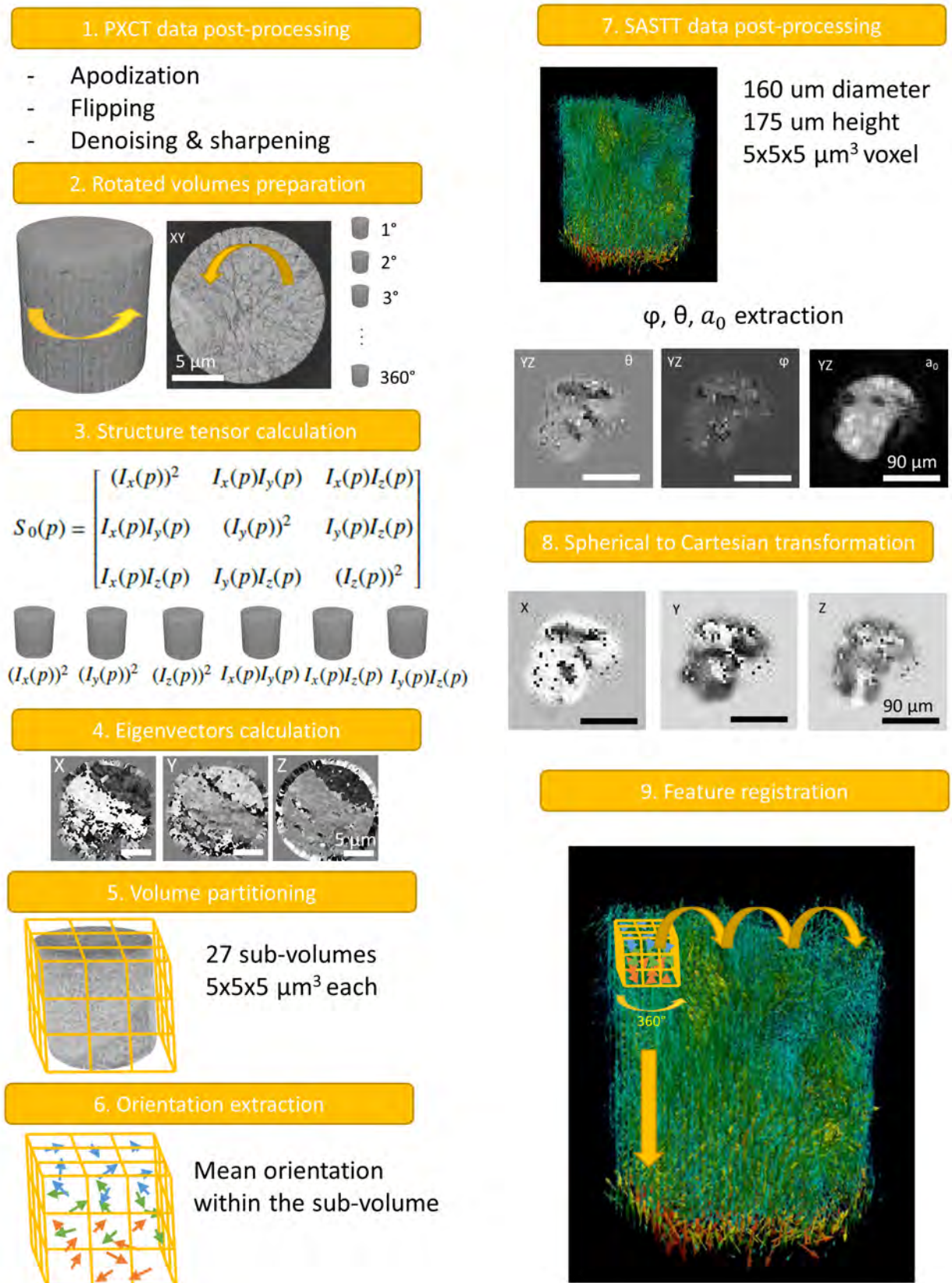
Figure 6: Schematic of the 3D structure tensor analysis pipeline.

volume corresponds to a 3x3x3 sub-volume of SASTT tensorial voxels.

Flipping. Data acquisition of SASTT and PXCT is performed in a different coordinate systems. Therefore, as the next step, the PXCT volume was flipped with respect to the x-axis to match SASTT coordinate system.

Denoising and sharpening. The next consideration in the pipeline was to attempt enhancing the image by filtering out noise or by sharpening it. The ptychographic reconstruction already undergoes a post-processing steps before the tomographic reconstruction (Guizar-Sicairos et al., 2011), in which issues like ramp removal and smoothing are addressed. Still, an attempt was made at improving the image quality after this routine by implementing noise filtering through anisotropic diffusion and a sharpening routine was developed with the hope of improving the definition of the 3D structures of interest. An improvement in image quality was not observed after the aforementioned filtering trials.

3.4.1.2. Rotated volumes preparation.
In this work, the feature registration task required to not only find the 3D spatial location of one volume within another but also the angular rotation around the tomographic axis. To this end, a set of rotated PXCT volumes were computed by spline interpolation in the angular range from 0 to 360° with the angular step of 1°, as illustrated in Figure 6, step 2.

3.4.1.3. Structure tensor calculation.
Also referred to as the second-moment matrix, the structure tensor matrix (presented in Equation 4 for a 3D case) is derived from the gradient of a function, in the case at hand, the image function. It describes the distribution of the gradient in a specified neighborhood around a point. In this work, the structure tensor is calculated for a 3-pixel wide neighborhood around every voxel of the ptychographic volume in all 3 directions. The result of this process were the 6 volumes illustrated in the 3rd step of Figure 6.

$$S(p) = \begin{bmatrix} (I_x(p))^2 & I_x(p)I_y(p) & I_x(p)I_z(p) \\ I_x(p)I_y(p) & (I_y(p))^2 & I_y(p)I_z(p) \\ I_x(p)I_z(p) & I_y(p)I_z(p) & (I_z(p))^2 \end{bmatrix} \quad (4)$$

where I_x, I_y, I_z are the three partial derivatives of the image neighborhood I, and the integral ranges over $\mathbb{R}^3$.

3.4.1.4. Eigenvectors calculation.
The eigenvectors $\hat{e}_1, \hat{e}_2, \hat{e}_3$, and the corresponding eigenvalues $\lambda_1, \lambda_2, \lambda_3$ of $S_w[p]$, synthesize the distribution of gradient directions in the neighborhood of p defined by the window w. In order to better visualize this information, it is often modelled as an ellipsoid whose semi-axes directed along the eigenvectors and their norms are equal to the eigenvalues.

The nature of the data set on which this analysis is performed is such that collagen fibrils and bundles of fibrils of different sizes are the structures of interest. In the case of fibrilar structures, the ellipsoid would correspond to a flat and round, disk-like shape. If λ_3 is much smaller than both λ_1 and λ_2, the gradient directions are spread out and perpendicular to e_3 (see supplementary figure 2b); so that the isosurfaces tend to be like tubes parallel to that vector. This situation occurs, for instance, when p lies on a thin line-like feature, or on a sharp corner of the boundary between two regions with contrasting values.

In the regions with very little to no organization, the gradients are dispersed in all directions so that the ellipsoid is roughly spherical ($\lambda_1 \approx \lambda_2 \approx \lambda_3$) (see supplementary figure 2c), showing that the gradient directions in the window are more or less evenly distributed, with no particular alignment; so that the image function is mostly *isotropic* in that neighborhood. This is an indication of spherical symmetry in the neighborhood of p. It is particularly interesting, as the calculation of the proportion of such PXCT neighborhoods in a certain region could be correlated with the information about symmetry provided by SASTT. An extreme version of this case is when the ellipsoid, which is almost sphere-like, degenerates to a point. This is indicative of a lack of significant gradients in the window that is considered.

The manner in which the structure tensor analysis was implemented in this project is multi-scale (see Equation 5) in the sense that the scale at which this tensor is calculated can vary. The multi-scale structure tensor (or multi-scale second moment matrix) of a function I is an image descriptor that is defined over two scale parameters. One scale parameter referred to as local scale t, is needed for determining the amount of pre-smoothing when computing the image gradient $(\nabla I)(x; t)$. Another scale parameter referred to as integration scale s, is needed for specifying the spatial extent of the window function $w(\xi; s)$ that determines the weights for the region in space over which the components of the outer product of the gradient $(\nabla I)(\nabla I)^{\text{T}}$ are accumulated. In the current algorithm, the window size was set to two, which was proposed as default for the fiber orientation analysis in the Püspöki et al. (2016) OrientationJ software. In consequence, the neighborhood is a 5x5x5 voxels[3] around the current voxel.

$$\mu(x; t, s) = \int_{\xi \in \mathbb{R}^k} (\nabla I)(x - \xi; t)(\nabla I)^{\text{T}}(x - \xi; t) w(\xi; s) d\xi$$
$$(5)$$

where t is a variable expressing the local scale that controls the pre-smoothing applied before calculating the gradient; $\nabla I(x; t)$ is the gradient along the x direction; the scale at which the integration takes place is
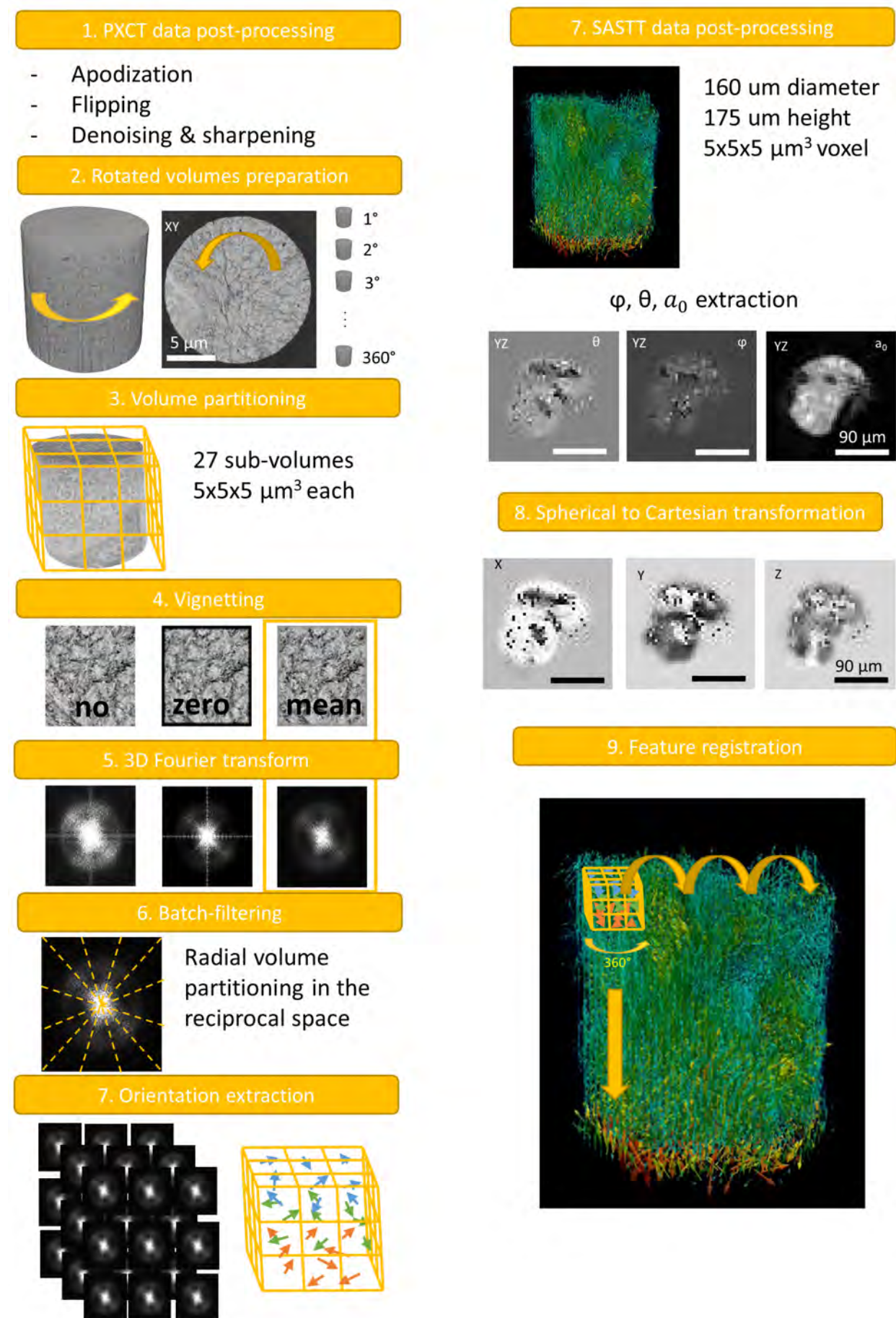
Figure 7: Schematic of the Fourier spectrum analysis pipeline.

expressed by s and the window that is considered is sig- nified by the window function $w(\xi; s)$.

$$det(\alpha I - S(p)) = \alpha^3 - \alpha^2 tr(S(p)) - \alpha\frac{1}{2}(tr(S(p)^2 - tr^2(S(p))) - det(S(p)) = 0 \qquad (6)$$

where α is a scalar, I is the 3x3 identity matrix, $tr(S(p))$ is the trace of the 3x3 second moment matrix, $S(p)$, and $det(S(p))$ is its determinant.

For the actual computation of the eigenvectors and eigenvalues, Smith's algorithm was implemented (Smith, 1961). This method starts off with the characteristic equation (Equation 6) of the 3x3 matrix that is the structure tensor that was calculated at the previous step for every voxel.

In Smith's implementation it is chosen to apply an affine change to S in order to simplify the expression $S = pB + qI$. In this expression, S and B have the same eigenvectors and an eigenvalue β of B exists if and only if $\alpha = p\beta + q$ is an eigenvalue of S. If we express q as $q = tr(S)/3$ and p as $p = \sqrt{tr((S - qi)^2)/6}$, then:

$$det(\beta I - B) = \beta^3 - 3\beta - det(B) = 0. \qquad (7)$$

By doing so, the computationally expensive solution of the equation through the Lagrange method is avoided and a trigonometric solution is reached.

After the substitution $\beta = 2cos\theta$ and the simplification by use of $cos3\theta = 4cos^3\theta - 3cos\theta$, Equation 7 becomes:

$$\beta = 2cos\left(\frac{1}{3}arccos\left(\frac{det(B)}{2}\right) + \frac{2k\pi}{3}\right), k = 0, 1, 2. \qquad (8)$$

Once the eigenvalues are computed, the eigenvectors are obtained as such: if λ_1, λ_2, λ_3 are the distinct eigenvalues of the structure tensor $S(p)$, then $(S(p) - \lambda_1 I)(S(p) - \lambda_2 I)(S(p) - \lambda_3 I) = 0$. In the general case, the columns of the product of any two of these matrices contains an eigenvector for the third eigenvalue. In the case in which two eigenvalues are identical, for example $\lambda_2 = \lambda_3$, then $(S(p) - \lambda_1 I)(S(p) - \lambda_2 I)^2 = 0$ and $(S(p) - \lambda_2 I)^2(S(p) - \lambda_1 I) = 0$, resulting in the generalized eigenspace of λ_2 to be spanned by the columns of $(S(p) - \lambda_1 I)$ and its ordinary eigenspace to be spanned by $(S(p) - \lambda_2 I)(S(p) - \lambda_1 I)$, while the eigenspace of λ_1 is spanned by $(S(p) - \lambda_2 I)^2$.

In the framework that was developed during the course of this project, the result of this step included 12 volumes: 3 arrays corresponding to the three eigenvalues and 9 arrays of the x, y, z coordinates of the three corresponding eigenvectors. The eigenvector used for the calculation of orientation was the one with the highest eigenvalue, as it is the one that indicates the main orientation of its neighborhood. Slices of the volumes generated by calculating the Cartesian coordinates of the biggest eigenvector are presented in step 4 of Figure 6.

3.4.1.5. Volume partitioning. As mentioned earlier in the pipeline, the remodelling of the PXCT volume allowed for its equivalence with a 3x3x3 region of the SASTT tensorial volume from dimensionality point of view. After the extraction of the three eigenvectors and three eigenvalues, each of the 12 volumes resulting from the *Eigenvectors calculation* step have been divided into 27 equally-sized sub-volumes (step 5 of Figure 6). One sub-volume is made of 250x250x250 voxel^3 and measures 125 μm^3.

3.4.1.6. Orientation extraction. The local orientation calculated around the neighborhood of each voxel in the initial PXCT volume is now averaged over the range of each sub-volume. This operation results in three 3x3x3 volumes for each of the three eigenvectors, expressing their orientation in the three Cartesian coordinates. Step 6 of Figure 6 symbolizes a visual rendering the information contained within the three volumes that correspond to each eigenvector. Hence, at the end of this step, nine 3x3x3 volumes are produced.

3.4.1.7. SASTT data post-processing. The exploration step for the SASTT data was of utmost importance, as it allowed for the proper extraction and handling of the features outputted by the *spherical harmonics*-based reconstruction. As outlined in previous chapters the features, that expresses the average local orientation over a window of 5x5x5 μm^3, in the case of SASTT is in spherical coordinates, represented by the ϕ and θ angles. The features that express the local anisotropy of the sample encode it in the form of spherical harmonic coefficients, a_0, a_1, a_2 and a_3.

3.4.1.8. Spherical to Cartesian transformation. The orientation information contained in SASTT volume had to be transferred to the same coordinate system as the one in which the orientation of the PXCT data was encoded in, the Cartesian system. This was achieved by applying the three transformations in Equation 1.

3.4.1.9. Feature registration. Following the feature extraction step from both data sets presented in this work, a correlation routine was developed. The SASTT volume was used as the main array and the PXCT orientation array was used as a correlation kernel. In other words, the 3x3x3x3 volume which contains information about the orientation of PXCT sub-volumes was slid

over the 3x43x48x48x48 SASTT volume of the Cartesian components of orientation for each tomographic angle as illustrated in the step 9 of Figure 6.

The manner in which the operation was set up was chosen such that the output array has the same dimensions as the bigger array, in our case, SASTT.

The correlation is computed for each component of the PXCT orientation array and its SASTT counterpart. Hence, the volume containing information about the X components of the eigenvectors extracted from PXCT is correlated with the X components derived from the ϕ and θ angles of SASTT. The same applies in the case of the Y and Z components. After the correlations of the three Cartesian components of a new array are computed, these are then multiplied, thus outputting an array in which the highest values are assigned to voxels in which the three components are in agreement.

3.4.2. Fourier spectrum analysis

The Fourier transform converts the input signal from the initial domain (real space) to its representation in the frequency domain (reciprocal space). The Fourier transform of an image can provide information about the objects as well as its texture. In this image processing framework based on the Fourier transform, the PXCT that was used in this work is treated as a 3D function of finite duration and its frequency components refer to the spatial frequencies present in the volume:

$$F(u,v,w) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1}\sum_{w=0}^{P-1} f(x,y)e^{-2j\pi(ux/M+vy/N+wz/P)}$$

(9)

where the function $f(x,y)$ is the tomographic volume with the size $MxNxP$, evaluated for the values of the discrete variables u, v, w in the ranges $u = 0, 1, 2, ..., M-1$, $v = 0, 1, 2, ..., N-1$, $w = 0, 1, 2, ..., P-1$.

3.4.2.1. PXCT data post-processing. For this method, the ptychographic volume underwent the same *Apodization, Cropping, Padding* and *Flipping* routines as for the structure tensor analysis approach.

3.4.2.2. Rotated volumes preparation. The volume also had to be rotated around the tomographic axis to prepare a set of rotated volumes. The rest of the processing in this pipeline was applied to the volume generated at each of these rotations, in the same manner as before.

3.4.2.3. Volume partitioning. The processing done in this pipeline begins to differentiate starting with the fact that the volume is partitioned prior to the rest of the processing (Figure 7, step 3).

3.4.2.4. Vignetting. Before performing the 3D Fourier transformation of the sub-volumes, 3D vignetting was applied to each of them in order to remove artifacts present in the reciprocal space due to the edges of the region of interest within the array.

The edges present within the PXCT volume are of two kinds: the ones generated by the sub-volume partitioning and the ones that are the consequence of the cylindrical geometry of the bone sample. Figure 7, step 3 helps with the envisioning of this two types of array edges.

Two kinds of vignetting were tested. First, a half-sine vignetting was applied to the image, resulting on a dimming towards zero of the image at the periphery of the region of interest.

The second method is more complex: instead of dimming the edges of the image to zero, the function that smooths out the edges of the image array is a dampened sinusoidal that stops its fluctuations at the mean value of the PXCT array.

3.4.2.5. 3D Fourier transform. The algorithm used to obtain the Fourier transform of the ptychographic volume was the fast Fourier transform (FFT) proposed by Cooley et al. (1969). This approach was chosen due to the computational efficiency that it provides, which was crucial in the development of this pipeline. Such speed of calculation is achieved through clever consideration of symmetries in the calculated terms, bypassing the need to calculate the transform by evaluating the transform in all the initial points. This algorithm allows for the Fourier transform to be computed almost 2200 times faster than by using the direct Fourier transform (DFT) on the same computational system (Rafael C. Gonzalez, 2008).

After transforming each volume to its reciprocal space, a shift was applied to the whole spectrum to make the first component appear in the center of the array and the modulus of the complex numbers was computed to obtain volumes such as the ones illustrated in Figure 7, step 5.

3.4.2.6. Orientation extraction. A few methods for extracting the orientation from the reciprocal (frequency) space were researched. The principal candidate for this pipeline is Gabor filtering. By using a batch of Gabor filter, it is possible to sample the information in the 3D Fourier space and, according to the values of each component of the catch of filters, infer the orientation of the collagen fibrils that make up the structures in the PXCT sub-volumes. This information is then expressed in 3D vectorial form in a manner similar to eigenvectors.

Another methodology that is considered for this step is the *principal component analysis* (PCA) of the Fourier space. This would result in a principal component vector that points in the direction of the main variation of data in the cloud of frequencies that make up

the reciprocal space of each of the 27 sub-volumes.

3.4.2.7. SASTT data post-processing. The manner in which SASTT data is extracted and transformed to fit the feature registration routine is the same as in the case of the aforementioned structure tensor analysis pipeline.

3.4.2.8. Feature registration. In the case of the Fourier spectrum analysis pipeline, the correlation routine would be exactly the same as the one proposed for the eigenvector analysis pipeline but instead of the PXCT orientation array coming from the main eigenvector of a sub-volume, it would come from the orientation vector perpendicular to the disc shapes created in the reciprocal space by the cylindrical aspect of the collagen fibrils.

3.4.3. Deep feature correlation

The final approach that was considered for this project was extracting the representation in *latent space* by training a 3D *autoencoder*-type convolutional neural network with the two data sets. An autoencoder is a kind of deep learning architecture that consists of two parts:

a) *The encoding path* is made up of the layers of the network that collectively learn how to best represent the input data in the latent space.

b) *The decoding path* which, for the purposes of this project, learns how to best reconstruct the input data.

The system is trained with the input and the output of the network being the same volume. Contrary to the common use of deep learning techniques, in this work it is only used as a mathematical tool for feature extraction, not for the generalization of performing a task on other data sets.

3.4.3.1. PXCT and SASTT data post-processing. As suggested in Figure 8, step 1, the PXCT data was only flipped along its X axis before inputting it in the network. The SASTT data only had to be extracted from the reconstruction files and, in this pipeline, information about anisotropy is also taken advantage of by using the first spherical harmonic coefficient (a_0) as shown in Figure 8, step 2.

Experiments were carried out by using both partitioned versions of the PXCT volume and the full PXCT volume as input to the autoencoder. For the sub-volume partitioning routines, data was fed to the model by means of *generator*-type routines. Since this time the correlation is not calculated between voxels, but between representations in latent space, it was considered useful to also let the network extract information from the volume as a whole. The same consideration applies to the SASTT data.

Prior to being inputted into the network, the arrays were modelled to dimensions that fit the TensorFlow (Abadi et al., 2015) framework. This translates to the

introduction of two new axes for both arrays and to the padding of SASTT data to dimensions compatible with the kernel size used for network operations.

3.4.3.2. Latent space representation. The result of the processing of the PXCT volume with the autoencoder neural network is one latent space vector, while in the case of processing the SASTT data, three volumes are passed through the network (θ, ϕ, a_0) so three latent space representations are extracted.

Optimization and hyper-parameters. The optimizer that was chosen for the training process is *Adam*. This algorithm is based on the adaptive estimates of the lower-order moments. Two parameters control the exponential decay of these moments: $beta_1$ which represents the exponential decay rate of the 1st moment, which in this project was set to 0.9; and $beta_2$, the exponential decay rate of the 2nd moment, in this work set to 0.999. ϵ is a small constant that serves the purpose of numerical stability. In this pipeline $\epsilon = 10^{-7}$. The learning rate is controlled throughout the training process by a scheduling routine that starts at 0.01 and decays with a rate of 0.8 in 100 steps. The cost function that was chosen for the model is the *mean squared error*.

Neural network architecture. The model that was used is represented by the diagram in Figure 8, step 3. The encoding of input data is done by means of 3D convolution and max pooling with steps of batch normalization between convolutions. The dimensionality of the layers becomes longer and thinner with each convolution operation. The number of filters on the input layer is two and it increases to four in the first convolutional layer. It stays four until the last convolution, where the number of filters is again two right before the output layer. The decoding of the latent space, and reconstruction of the input data, is realized by use of 3D convolutions as well, but instead of max pooling, up-sampling is used. In this way, the model learns how to populate an array with the original dimensionality at each of the five steps of the decoding path. The dimensionality of the kernels used throughout the network is 5x5x5 in the case of the PXCT volume and 3x3x3 in the case of the SASTT data.

After the model is trained, it is saved to disk and for the deep feature extraction, only the encoding path is imported and used to process the data by taking advantage of a routine that was developed for the automatic naming of layers: only the layers that are called by name are loaded in the predictor model. The output of the prediction function call is the latent space of the input data.

3.4.3.3. Sub-sampling of the latent space. The resulting latent space was just a 1D vector with a dimensionality of 864x1 in the case of PXCT and 48x1 in the case of the three SASTT volumes as depicted at the end of step 3 in Figure 8. Therefore, before the correlation routine that concludes the pipeline, the PXCT latent space
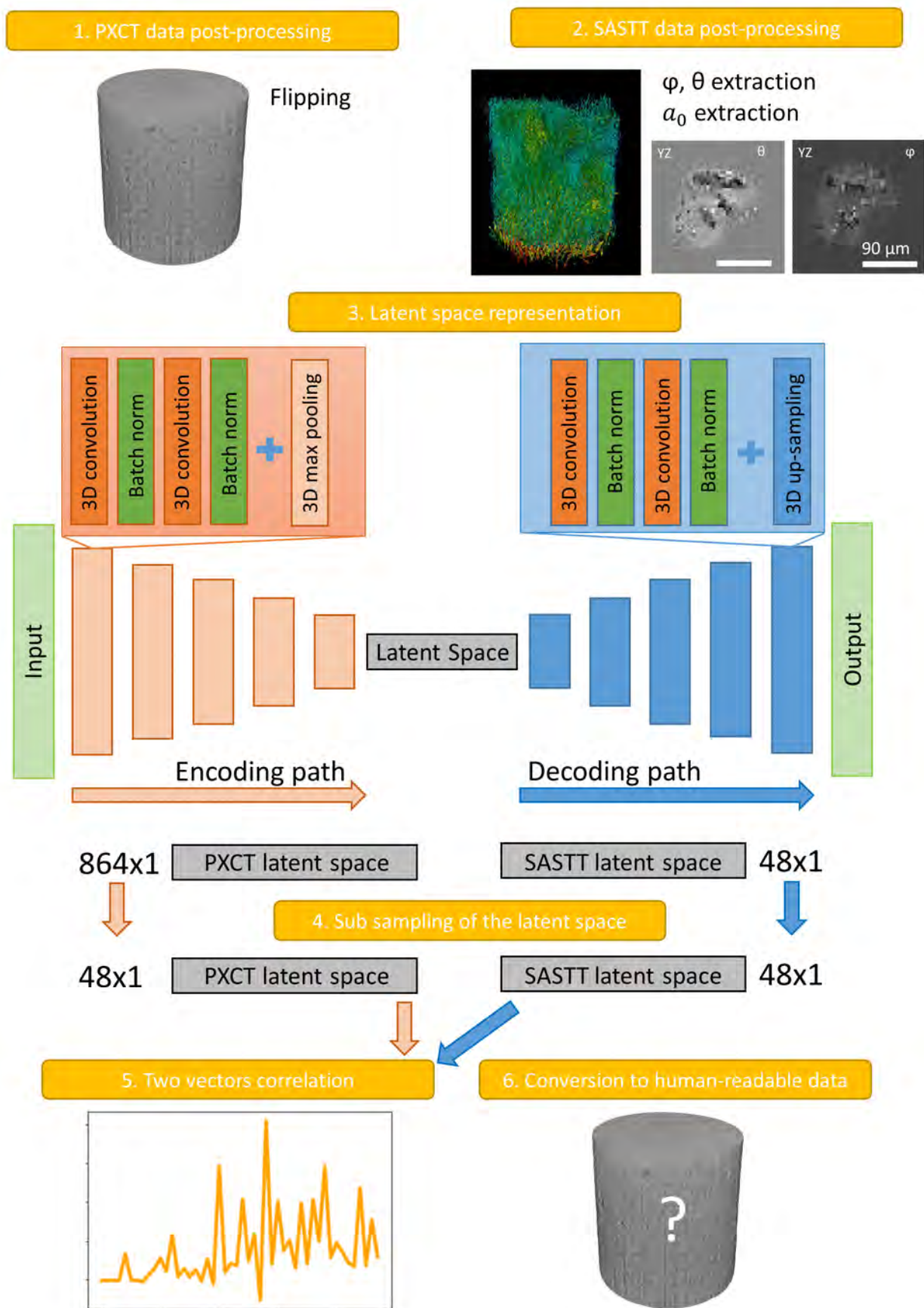
Figure 8: Schematic of the deep feature correlation pipeline.

vector is sub-sampled in order to reach the same dimension as the SASTT representation as suggested by the arrows in step 4 in Figure 8.

3.4.3.4. Two vectors correlation. The correlation is computed between the latent representation of PXCT and all three deep feature vectors of SASTT. The results of the three correlation are then multiplied, resulting in only one final correlation vector as illustrated in Figure 8, step 5.

3.4.3.5. Conversion to human-readable data. The final step of this pipeline is the conversion of the deep feature correlation result to a representation that is interpretable by human experts, see Figure 8, step 6. This was attempted by using the *decoding path* of the models that were used.

4. Results

4.1. Structure tensor analysis

The structure tensor analysis provided the principle results in this work. The strongest correlation between two data sets was found and a quantitative assessment of the correlation between the PXCT and SASTT features at each rotation is depicted in Figure 9 from various perspectives. At each angular step of the tomographic rotation, a correlation array was outputted. The curve depicted in graph b) presents the maximum value of every correlation array computed at every tomographic angle. In the same manner, graph c) depicts its minimum value and graph d) illustrates the average value of that correlation array.

This correlation routine outputs one array for each tomographic rotation as stated above. If this resulting array is flattened and plotted at each rotation, graph a) of Figure 9 is obtained. The values in graph a) are so tightly packed because instead of only one value for each rotation, the whole flattened array of correlation coefficients is squeezed into a very tight spacing on the *x* axis.

The highest correlation coefficient was found to have a value of 2.236 and is apparent in Figure 9a,b and d at a tomographic rotation of 89°. So not only is 2.236 the highest correlation coefficient but it also occurs within a range of rotations that present consistently high average correlation values. Other local maxima also do exist in Figure 9a,b. These locations in the data were explored but they corresponded to surface areas of the pillar so likely caused by sample preparation artefacts from material re-deposition.

The minimum correlation (apparent in Figure 9a,c) was found at 158°. Two very well defined negative peaks are present in Figure 9c between 200° and 360°. In Figure 9d depicting the average correlation coefficients, there is only one negative correlation peak at 313°. These regions were also explored, mainly to rule
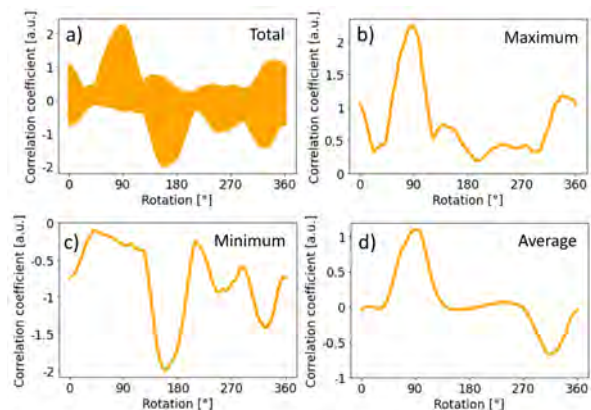

Figure 9: Results of the 3D structure tensor analysis: a) total correlation, b) maximum correlation, c) minimum correlation, d) average correlation. The clear correlation coefficient maximum is present at 89° at the spatial coordinates of x=21, y=23, and z=34 px.

out the algorithm misbehaviour, i.e. to establish that the resulting correlation volumes still resemble the pillar shape.

A correlation array has the size of a SASTT volume as it is a result of *scanning* the SASTT feature volumes, that have a dimensionality of 48x48x43 with the 3x3x3 PXCT feature volumes at a given tomographic rotation. The correlation array which contained the highest correlation values, at the above mentioned 89° rotation is presented in Figure 10. In the first image, a), the XY plane is presented and in the second one, b), the correlation array is depicted in the YZ plane. Note that Figure 10a resembles the axial cut of the SASTT pillar, while the Figure 10b corresponds to the sagittal cut of the SASTT pillar. It is clear that the location of the highest correlation coefficient is present close to the top pillar surface, roughly in the center of the pillar.

The lines overlaid on the images of Figure 10a,b represent the spatial 1D ranges over which the line profiles in c, d and e were computed. These profiles put the highest correlation value in quantitative perspective, giving insight into its signal-to-noise ratio with respect to the rest of the correlation coefficients on the same longitudinal, latitudinal and vertical level as this voxel. The overlays are color coded so that each of them correspond to the line profile of the some color.

The voxel with the highest correlation is located in the central region of the very top of the bone pillar. In the line profile graphs it appears as a peak with relatively high values around it. The position of this voxel reveals the position of a 3x3x3 ($15x15x15\,\mu m^3$) sub-volume of the SASTT tensorial volume that is indicated to correspond to the PXCT volume of the same dimensions.

4.2. Deep features correlation

Figure 11 illustrates a plot of the correlation vector that resulted from the deep feature analysis pipeline. The distribution of values in this vector presents a heavy
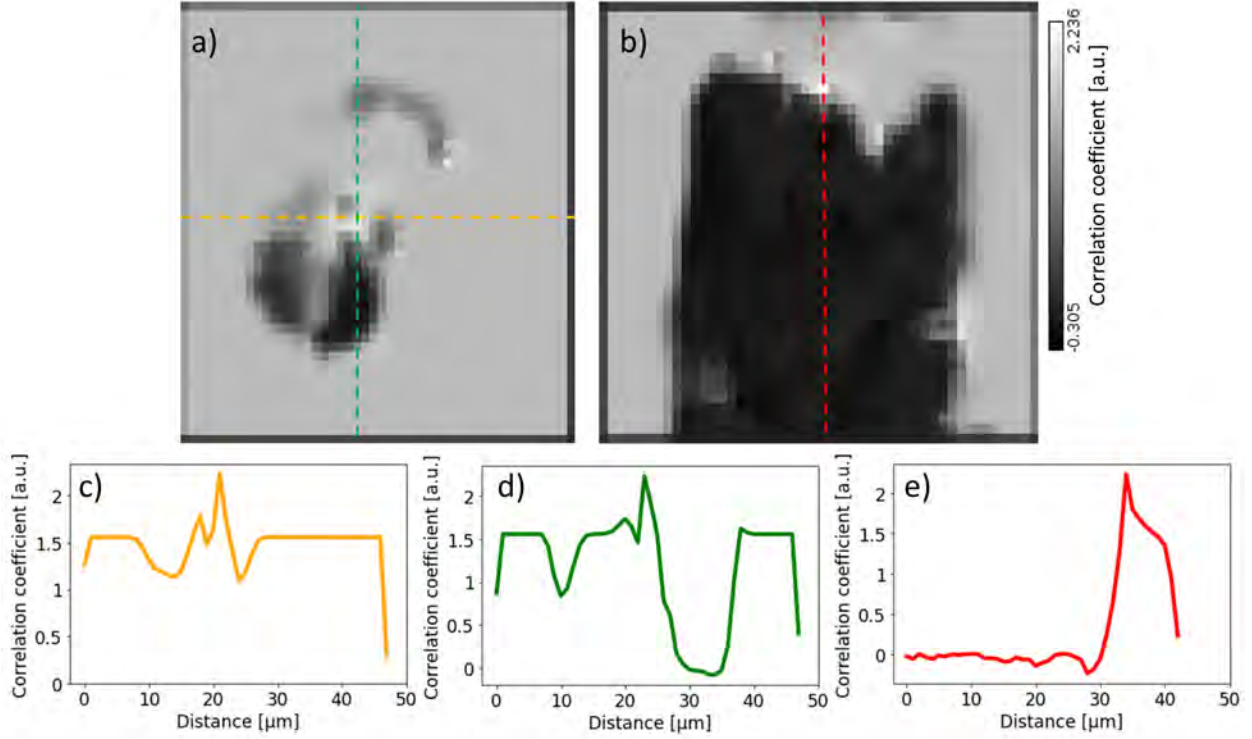
Figure 10: Results of the 3D structure tensor correlation: a) the axial plane, b) the sagittal plane of the correlation coefficient map at 89°. The bright regions show the highest correlation. The line profiles in three directions are presented in c), d) and e).

right tail, the whole array being skewed towards that direction. The values of the vector start from zero and then grow in a rough manner until the maximum is reached then they decrease significantly but never return to zero again.

The aspect of this correlation array is jagged, in opposition to the relatively smooth line profiles that resulted from the structure tensor analysis, depicted in Figure 10. The maximum correlation is found to be at the index of 28 of the resulting latent space vector. However, it is needed to convert this data back into human-readable format to draw conclusions on where in the SASTT 3D volume this correlation is found and at which rotation angle.

5. Discussion

5.1. Results interpretation

As stated in the *Materials and methods*, the PXCT volume was collected from a pillar of bone that was previously explored with SASTT. The sample was decreased in dimensions from 160 μm to 15 μm in diameter to fit PXCT requirements. By doing so the effort was taken to know the location of the newly obtained PXCT volume and this location was at the very top of the SASTT pillar, in the center. The only unknown factor is the rotation of one pillar with respect to another as this was not preserved between two measurements.
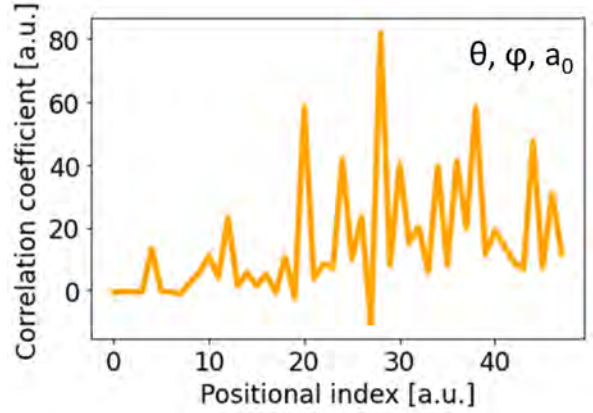
Figure 11: Results of the deep feature correlation based on the θ and ϕ angles and the principal component a_0. The maximum correlation is found to be at the index of 28 of the latent space vector.

The results obtained by 3D structure tensor correlation in this work correspond well to the prior knowledge of the location of PXCT pillar in the SASTT volume. In Figure 10 it is clearly depicted that the voxel with the highest correlation coefficient is located at the top of the SASTT pillar, and that it is also central. Moreover, this voxel has the higher value than any other voxel in all correlation arrays at all orientations. Taking all of these arguments into consideration, the results point towards the fact that these findings represent a form of validation for the aforementioned feature correlation pipeline.

Even though the Fourier spectrum analysis pipeline is not complete and require further development, the results of the Fourier transform routine (Figure 7, step 5) clearly illustrate a very important detail: within one sub-volume of the PXCT data, multiple fiber orientations can be distinguished. It can be seen from the fact that in the reciprocal space representation there are two or more intersecting regions of high intensity. In the presented slice in Figure 7, step 5, one of them is at 80°-85° and the other one at around 135°. These correspond to two ellipsoidal shapes in 3D that intersect, corresponding to the two orientation present within the region that was Fourier transformed. The ellipsoidal shape of these transforms comes from the fact that the Fourier transform of a thin cylinder (i.e. *fiber*) is an ellipsoid. This means that the Fourier spectrum analysis pipeline once complete will be able to work with and incorporate more than one collective orientations and therefore to describe the structures with higher precision. This can not be achieved with the 3D structure tensor correlation pipeline.

It is important to note that the deep feature correlation pipeline was able to find a strong correlation as well, but the data remains to be converted to the human-readable format for drawing conclusions and validation. Even thought it requires further developments, this method would be most versatile and flexible to accommodate different feature-rich types of data from variety of experimental methods, which can not be done with previously mentioned pipelines.

5.2. Algorithmic considerations

In the incipient stages of this project, while performing the literature review, an exploratory phase took place. During this exploration, a plethora of methods were considered for accomplishing the main task of this project. Out of these methods, the three pipelines that are presented in this work began to take shape. The Fourier spectrum analysis (FSA), the structure tensor analysis (STA) and the deep feature correlation (DFC) pipelines were developed in parallel for the most part. As the intermediary results of STA looked promising and the experiments within this framework were repeatable, all efforts were shifted towards this approach. This decision made it the algorithmic framework of choice for solving the proposed task. This leads to the fact that most of the results come from the structure tensor analysis approach.

There are, of course, other considerations for which the STA pipeline was chosen. First of all, the mathematical basis upon which it was constructed is very transparent and explainable. Moreover, it is a tried and tested method in the imaging community, as outlined in Section 2. *State of the art*, giving consistent and robust results for the analysis of orientation in data coming from different imaging modalities. Last but not least, this

method can be applied natively in 3D, a very important condition for the project.

Final results of the Fourier spectrum analysis are not included in this work as the development efforts were shifted mainly towards STA. One output of this pipeline that aided STA was the transformation of PXCT grayscale sub-volumes. Apart from providing further qualitative insight into the multi-directional nature of collagen fibers within one sub-volume, this served as a qualitative assessment of performance during anisotropic diffusion and sharpening experiments.

The DFA approach yielded interesting results but further research is needed into how to express them in a more comprehensive manner. It is noteworthy that this was the only framework in which features that encode information about the local anisotropy of the bone sample were also included in the correlation routine.

5.3. Computation

All the computations carried out throughout this project was performed on the *Ra* cluster of the Paul Scherrer Institute. This is a high-power super-computing cluster of CPU and GPU nodes directed towards *offline* data analysis. In this context, the term "offline analysis" refers to the processing of data that happens after an experiment that was performed at the synchrotron.

The main challenge in the case of DFA was memory management. This framework was developed for GPU processing. As it was deemed useful to train the autoencoder model with the full 3D volume, a lot of VRAM (video random access memory) was required to operate with the whole array. This resulted in the quota being routinely reached so memory cleaning routines were implemented in the code.

The calculations from the current stable version of the STA pipeline are performed on CPU. This aspect makes the time required to run it relatively high, at around 72 hours. A GPU-based version of this pipeline was also developed. Its usage can help reduce the time required for the feature extraction and registration to under 4 hours but further testing is required in order for it to be deemed robust enough.

The calculations that the FSA pipeline is based upon all run on CPU and from a time perspective, the time required to perform them is under one hour.

For all the computation that required the use of shared resources, routines were developed and implemented that check the status of available computational nodes and help the user only select those which are not currently and use.

5.4. Future work

The pipelines developed during the course of this project can benefit from a number of further implementations that have been considered but which were not applied mainly due to time constraints.

Structure tensor analysis. Currently, the features that are used for registration are only those that encode orientation. One improvement could be the inclusion of features that encode symmetry and anisotropy. From the SASTT side, these are the spherical harmonics coefficients which have already been extracted from the reconstruction array. In the case of PXCT, one more step of processing needs to be carried out, the processing of the eigenvalues volumes. Aside from the inclusion of new features that can help the registration, improvements can be made to this routine by means of optimizing the partitioning of the PXCT volume. There is no way of guaranteeing that the current partitioning of the PXCT volume is the optimal one. To explore this, the 3D gridding that separates the sub-volumes can be optimized.

Fourier spectrum analysis. Fully developing this pipeline can provide further validation of the results found by structure tensor analysis. Moreover, since the reciprocal space contains several ellipsoids in the transform of only one PXCT sub-volume, multiple orientations could be extracted and quantified. The next step is to find a suitable method for the extraction of orientation information from the computed Fourier space and to use it to obtain a vectorial representation of this information for correlation with the SASTT data. One such technique is extracting the principal components of the reciprocal space data by means of PCA. Another potential idea for the task of feature extraction is the processing of the Fourier space with a batch of Gabor filters.

Deep feature analysis. More research is needed for finding a way to make the latent space correlation vector interpretable by humans. While the correlation array that results from this pipeline does not appear to be fully random, it is certainly not a good representation of feature registration from a human interpretability point of view. Moreover, a more thorough design work for the architecture can be done, resulting in latent space dimensionality that would not require further sub-sampling after the feature extraction step.

5.5. Other considerations

From a human interpretability point-of-view, the STA and FSA pipelines output the best results as opposed to DFA. However, DFA outperforms the two previous pipelines in terms of scalability, this means that DFA is easily applicable to other experimental data without major pipeline modifications. In addition, little to no pre-processing was required for the DFA implementation, while STA and FSA required some data manipulation in terms of filtering and interpolation. In the case of STA, for example, the gridding strategy has to be changed to accommodate a new data set, by contrast DFA does not require this partitioning step.

One other consideration that has to be made when comparing the three pipelines is the space required by each. For STA and FSA, a large amount of intermediary

volumes were saved in order to save computational time in later stages of development. This was not the case for DFA which, due to its lack of required pre-processing steps it did not require intermediary data to be saved to disk.

As stated previously, one consideration for adopting STA as the main computational framework of this project is the community's trust towards the mathematical mechanisms upon which it is based. The same argument can apply for FSA because the main routine in this framework is the fast Fourier transform which is widely accepted as a standard engineering and scientific analysis tool. This cannot be said about DFA as the majority of the scientific community still considers data-driven approaches to be "black boxes". On the other hand, the DFA approach is very novel and can attract high-impact applications.

Ultimately, it would be great to compare the performance and to cross-validate the three described pipelines as well as test their performance on the multiple samples available in the project.

6. Conclusions

In this work we present an algorithmic framework for the registration of two tomographic data sets acquired with two different state-of-the-art methods: ptychographic X-ray computed tomography and small-angle scattering X-ray tensor tomography. This task was accomplished not only spatially in the 3 Cartesian directions, but also the rotation of one volume with respect to another around its tomographic axis was found. Additionally, the two data sets were converted to contain comparable features, i.e. fibrillar orientations, on which the registration routine was based. The principal results were obtained by means of structure tensor analysis. This was the first time such a correlative study was performed and the results confirm the *a priori* considerations of the project, from sample preparation point of view, validating the SASTT reconstruction and the physical meaning of a tensor in SASTT. The two other proposed methods, Fourier spectrum analysis and deep feature analysis, require further development, but the ideas were proposed to how to tackle the necessary steps.

It is important to conclude that the PXCT and SASTT are complimentary techniques. SASTT provides very large field of view and offers 3D orientation information from large sample volumes, while PXCT offers very high resolution (down to 20 nm in biological tissues) and direct imaging. This work shows how by harnessing the power of both methods, not only the better fundamental understanding of bone is achieved but also new insights can be gathered by studying the effects of species, age, gender, pathologies on the collagen fibrils orientation in bone. Such an orientation-based feature

Feature registration algorithms for the correlative study of bone mineralized fibrils with small-angle scattering tensor tomography and ptychographic X-ray computed tomography

19

registration framework can also be useful to other disciplines such as material science and geology by providing the useful means of correlating orientation-specific data acquisition techniques with imaging modalities that output data in the real space.

Acknowledgments

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. software available from tensorflow.org.

Cooley, J., Lewis, P., Welch, P., 1969. The finite fourier transform. IEEE Transactions on Audio and Electroacoustics 17, 77–85. doi:10.1109/TAU.1969.1162036.

Diaz, A., Trtik, P., Guizar-Sicairos, M., Menzel, A., Thibault, P., Bunk, O., 2012. Quantitative x-ray phase nanotomography. Physical Review B - Condensed Matter and Materials Physics 85. doi:10.1103/PhysRevB.85.020104.

Dierolf, M., Menzel, A., Thibault, P., Schneider, P., Kewish, C.M., Wepf, R., Bunk, O., Pfeiffer, F., 2010. Ptychographic x-ray computed tomography at the nanoscale. Nature 467, 436–439. doi:10.1038/nature09419.

Faulkner, H.M., Rodenburg, J.M., 2004. Movable aperture lensless transmission microscopy: A novel phase retrieval algorithm. Physical Review Letters 93. doi:10.1103/PhysRevLett.93.023903.

Georgiadis, M., Müller, R., Schneider, P., 2016. Techniques to assess bone ultrastructure organization: Orientation and arrangement of mineralized collagen fibrils. Journal of the Royal Society Interface 13. doi:10.1098/rsif.2016.0088.

Guizar-Sicairos, M., Boon, J.J., Mader, K., Diaz, A., Menzel, A., Bunk, O., 2015. Quantitative interior x-ray nanotomography by a hybrid imaging technique. Optica 2, 259. doi:10.1364/optica.2.000259.

Guizar-Sicairos, M., Diaz, A., Holler, M., Lucas, M.S., Menzel, A., Wepf, R.A., Bunk, O., 2011. Phase tomography from x-ray coherent diffractive imaging projections.

Guizar-Sicairos, M., Georgiadis, M., Liebi, M., 2020. Validation study of small-angle x-ray scattering tensor tomography. Journal of Synchrotron Radiation 27, 779–787. doi:10.1107/S1600577520003860.

Heel, M.V., Schatz, M., 2005. Fourier shell correlation threshold criteria. Journal of Structural Biology 151, 250–262. doi:10.1016/j.jsb.2005.05.009.

Holler, M., Diaz, A., Guizar-Sicairos, M., Karvinen, P., Färm, E., Härkönen, E., Ritala, M., Menzel, A., Raabe, J., Bunk, O., 2014. X-ray ptychographic computed tomography at 16 nm isotropic 3d resolution. Scientific Reports 4. doi:10.1038/srep03857.

Holler, M., Ihli, J., Tsai, E.H., Nudelman, F., Verezhak, M., Berg, W.D.V.D., Shahmoradian, S.H., 2020. A lathe system for micrometre-sized cylindrical sample preparation at room and cryogenic temperatures. Journal of Synchrotron Radiation 27, 472–476. doi:10.1107/S1600577519017028.

Holler, M., Raabe, J., Diaz, A., Guizar-Sicairos, M., Wepf, R., Odstrcil, M., Shaik, F.R., Panneels, V., Menzel, A., Sarafimov, B., Maag, S., Wang, X., Thominet, V., Walther, H., Lachat, T., Vitins, M., Bunk, O., 2018. Omny - a tomography nano cryo stage. Review of Scientific Instruments 89. doi:10.1063/1.5020247.

Jiang, H., Ramunno-Johnson, D., Song, C., Amirbekian, B., Kohmura, Y., Nishino, Y., Takahashi, Y., Ishikawa, T., Miao, J., 2008. Nanoscale imaging of mineral crystals inside biological composite materials using x-ray diffraction microscopy. Physical Review Letters 100. doi:10.1103/PhysRevLett.100.038103.

Khan, A.R., Cornea, A., Leigland, L.A., Kohama, S.G., Jespersen, S.N., Kroenke, C.D., 2015. 3d structure tensor analysis of light microscopy data for validating diffusion mri. NeuroImage 111, 192–203. doi:10.1016/j.neuroimage.2015.01.061.

Kis, V.K., Czigány, Z., Dallos, Z., Nagy, D., Dódony, I., 2019. Hrtem study of individual bone apatite nanocrystals reveals symmetry reduction with respect to p63/m apatite. Materials Science and Engineering C 104. doi:10.1016/j.msec.2019.109966.

Liebi, M., Georgiadis, M., Kohlbrecher, J., Holler, M., Raabe, J., Usov, I., Menzel, A., Schneider, P., Bunk, O., Guizar-Sicairos, M., 2018. Small-angle x-ray scattering tensor tomography: Model of the three-dimensional reciprocal-space map, reconstruction algorithm and angular sampling requirements. Acta Crystallographica Section A: Foundations and Advances 74, 12–24. doi:10.1107/S205327331701614X.

Liebi, M., Georgiadis, M., Menzel, A., Schneider, P., Kohlbrecher, J., Bunk, O., Guizar-Sicairos, M., 2015. Nanostructure surveys of macroscopic specimens by small-angle scattering tensor tomography. Nature 527, 349–352. doi:10.1038/nature16056.

Püspöki, Z., Storath, M., Sage, D., Unser, M., 2016. Transforms

and operators for directional bioimage analysis: A survey. Advances in Anatomy Embryology and Cell Biology 219, 69–93. doi:10.1007/978 − 3 − 319 − 28549 − 8_3.

Rafael C. Gonzalez, R.E.W., 2008. Digital Image Processing.

Rezakhaniha, R., Agianniotis, A., Schrauwen, J.T., Griffa, A., Sage, D., Bouten, C.V., Vosse, F.N.V.D., Unser, M., Stergiopulos, N., 2012. Experimental investigation of collagen waviness and orientation in the arterial adventitia using confocal laser scanning microscopy. Biomechanics and Modeling in Mechanobiology 11, 461–473. doi:10.1007/s10237-011-0325-z.

Schmarje, L., Zelenka, C., Geisen, U., Glüer, C.C., Koch, R., 2019. 2D and 3D Segmentation of uncertain local collagen fiber orientations in SHG microscopy. URL: $http : //arxiv.org/abs/1907.12868http : //dx.doi.org/10.1007/978 − 3 − 030 − 33676 − 9_26$, doi:10.1007/978 − 3 − 030 − 33676 − 9_26.

Smith, O.K., 1961. Eigenvalues of a symmetric 3×3 matrix. Commun. ACM 4, 168. URL: https://doi.org/10.1145/355578.366316, doi:10.1145/355578.366316.

Thibault, P., Dierolf, M., Bunk, O., Menzel, A., Pfeiffer, F., 2009. Probe retrieval in ptychographic coherent diffractive imaging. Ultramicroscopy 109, 338–343. doi:10.1016/j.ultramic.2008.12.011.

Thibault, P., Guizar-Sicairos, M., 2012. Maximum-likelihood refinement for coherent diffractive imaging. New Journal of Physics 14. doi:10.1088/1367-2630/14/6/063004.

Verezhak, M., 2016. Multiscale characterization of bone mineral: new perspectives in structural imaging using X-ray and electron diffraction contrast.

Verezhak, M., Rauch, E.F., Véron, M., Lancelon-Pin, C., Putaux, J.L., Plazanet, M., Gourrier, A., 2018. Ultrafine heat-induced structural perturbations of bone mineral at the individual nanocrystal level. Acta Biomaterialia 73, 500–508. doi:10.1016/j.actbio.2018.04.004.

Weiner, S., Wagner, H.D., 1998. The material bone: Structure-mechanical function relations. Annu. Rev. Mater. Sci 28, 271–98. URL: www.annualreviews.org.

Feature registration algorithms for the correlative study of bone mineralized fibrils with small-angle scattering tensor tomography and ptychographic X-ray computed tomography

Alexandru-Petru Vasile, Mariana Verezhak

22.21
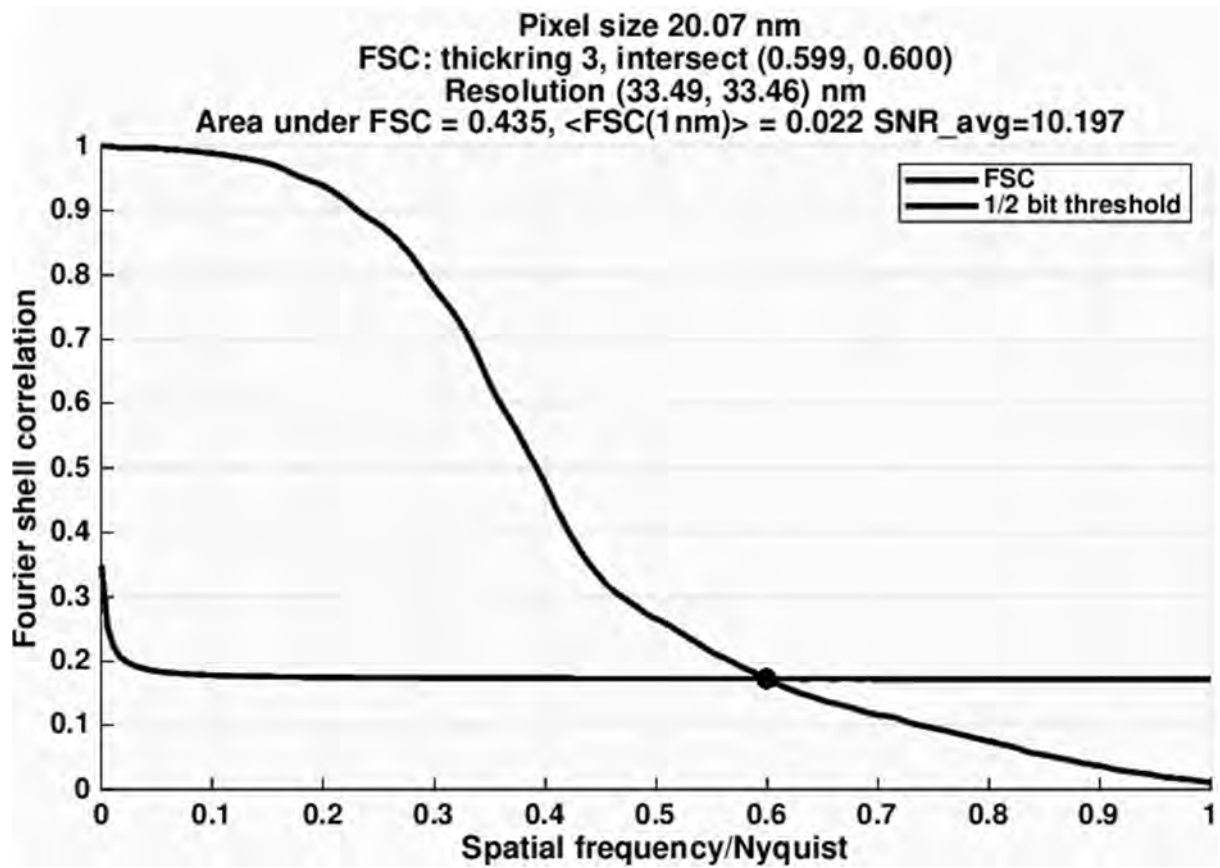
Supplementary figure 1: Resolution determined by the Fourier shell correlation curve at the intersection with the half-bit threshold.
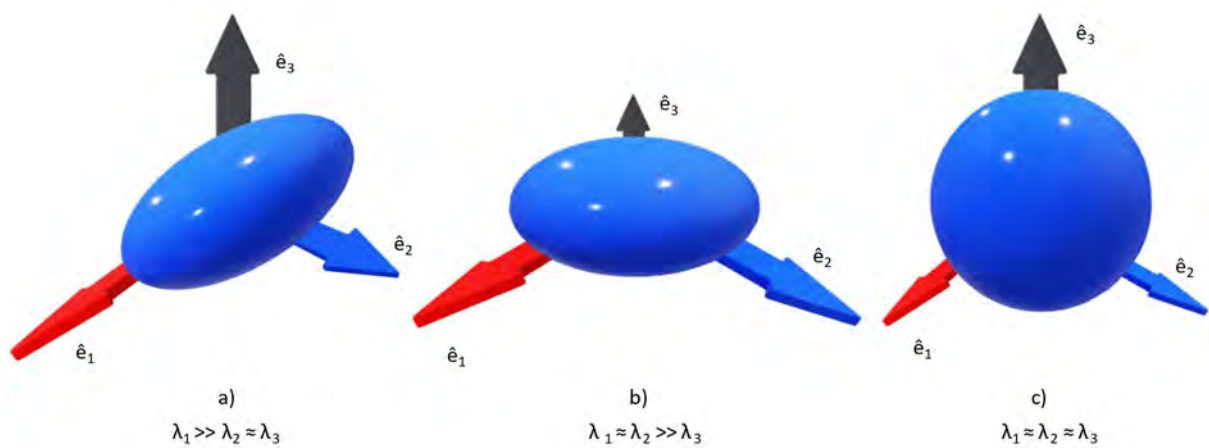
Supplementary figure 2: a) ellipsoid stretched along ê1 only; b) ellipsoid stretched in the direction of both ê1 and ê2; c) when all three eigenvalues are roughly equal, the ellipsoid becomes a sphere.
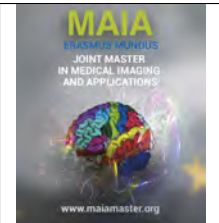
Knowledge-guided Segmentation of Isointense infant Brain

Jana Vujadinovic, Jaime Simarro Viana and Diana M. Sima

icometrix, Leuven (3000), Belgium

Abstract

Tissue segmentation of infants could lead to early diagnosis of different neurological disorders, potentially enabling early interventions. However, the highly dynamic developmental changes in the first year of brain development make tissue quantification challenging. The myelination, which progresses from central to peripheral brain regions, causes limited contrast between gray and white matter tissue on T1-weighted and T2-weighted magnetic resonance images at around six months. Previous studies usually base their work on implementing a well-known learning-based algorithm without utilizing auxiliary information to handle the problem. In this work, we propose a knowledge-guided U-Net for segmenting the isointense infant brain by considering the value of auxiliary anatomical information. Particularly, in one experiment, we adopt white matter prior obtained utilizing an available 6-month-old atlas to guide the segmentation. Conversely, in the second experiment, the low contrast boundary between gray and white matter is utilized as a second output channel to guide the segmentation in ambiguous regions. Experimental results on the subjects of the MICCAI Grand Challenge on 6-month infant Brain MRI Segmentation (iSEG19) challenge display the potential of utilizing the white matter prior as input for segmentation. Overall, more refined results and increase in segmentation accuracy was obtained when utilizing the prior compared to when not.

Keywords: guided segmentation, isointense phase, U-Net

1. Introduction

Pathologies related with the cognitive function (e.g. autism, intellectual disability) are mainly diagnosed after the appearance of clinical symptoms (Damiano et al. (2014), Lord et al. (2018)). However, early diagnosis of neurological disorders in children is highly relevant in the clinical practice since interventions have been proved to be more effective when they are provided at an early stage of development (Elder et al. (2017), Haefner and Maurer (2006)). Magnetic resonance imaging (MRI) allows the study of the brain in-vivo while neurological disorders could be identified before their onset by detecting brain imaging anomalies (Knickmeyer et al. (2008)). Specifically, MRI quantification of the brain tissues (i.e., white matter (WM) and grey matter (GM)) has been proved to be helpful for the early detection of neurological disorders such as schizophrenia (Gilmore et al. (2010)) and autism (Hazlett et al. (2005)). Furthermore, longitudinal measurements (i.e., repeated measurements over time on

the same individuals) enable a quantitative analysis of brain development, potentially leading to the prognosis of clinical outcomes.

Despite the potential clinical value of developing automatic methods in infants, MRI quantification of very young patients (i.e., from birth to 2 years old) presents multiple challenges. Infant MR scans suffer from lower quality as the result of increased partial volume effect due to smaller brain size, as well as motion artifacts (de Macedo Rodrigues et al. (2015)). In addition, rapid and non-linear neurodevelopmental changes contribute to heterogeneous intensities in MR scans leading to unclear borders and regional variations in contrast (Paus et al. (2001)). Furthermore, equipment manufacturers, magnetic field strength, and acquisition protocol can affect the contrast and intensity distribution in acquired images leading to multi-site heterogeneity issues (Sun et al. (2021)).

The highly dynamic developmental changes, especially in the first year of brain development, add further challenges to MR quantification. These developmen-

tal changes occur at micro and macro-structural levels. The myelination process induces an increase in lipids within the cell membrane, reducing the free water content. This accelerates the realignment of proton spins with the main magnetic field during MR image acquisition. Moreover, this myelination process is regionally dependent, as the oligodendrocyte myelination of subcortical and cortical regions develops in the posterior to anterior and caudal to the cranial direction (Zimmerman and Fuhrman (2011), Adam et al. (2014)).

Based on the distinct MR scan properties, these biochemical changes can be divided into three major phases: infantile phase, isointense phase, and adult-like phase (Paus et al. (2001)). As Figure 1 illustrates, during the infantile phase (samples of 2 weeks and 3 months), due to incomplete myelination of the brain, WM and GM intensities are reversed compared to adult brain imaging, which leads to hyperintense GM and hypointense WM in T1-weighted images. The pattern can be observed in the histogram, where GM intensities are higher than the intensities in WM. Furthermore, the T2-weighted images showcase higher contrast than the T1-weighted images. Over time, myelination progresses in the WM, which leads to age-related intensity changes. The isointense phase (sample of 6 months) corresponds to the phase in which the GM and WM exhibit similar intensities (in both T1-weighted and T2-weighted images), with a visible intensity overlap of the two classes present in the histogram. In the adult-like phase (samples of 9-12 months), on the T1-weighted images, the GM demonstrates hypointense intensities and WM hyperintense, similar to what is seen in normal adult brains. Concurrently, on the T2-weighted images, GM demonstrates hyperintense intensities and WM hypointense. In the histogram of T1-weighted images, this leads to WM taking on higher-intensity voxels and GM falling in the range of lower intensities. In the T1-weighted images, cerebrospinal fluid (CSF) has the lowest intensities in each phase and can be easily distinguished from the other two tissue classes, whereas in the T2-weighted images, it has the highest intensities.
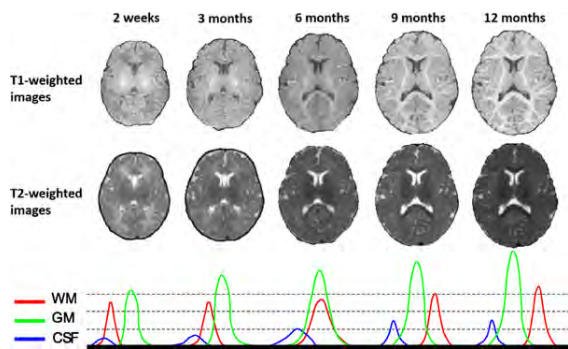

Figure 1: Visual presentation of brain MRI at different age stages during infant development. The respective histograms are result of T1-weighted image intensities. Courtesy of Wang et al. (2019)

iSeg17 and iSeg19 are publicly available MRI datasets focusing on infants in the isointense phase (Sun et al. (2021), Wang et al. (2019)). iSeg19 provides data points from different acquisition sites (Sun et al. (2021)). Other datasets such as Baby Connectome project (Howell et al. (2019)) and the dataset described in de Macedo Rodrigues et al. (2015) provide a wider range of ages, but unfortunately with limited access.

Given the necessity to develop a reliable approach for segmenting isointense brains and potentially assisting in the early diagnosis, the main aim of this research is to develop a robust method for isointense brain tissue quantification.

2. State of the art

Due to its clinical relevance, MRI quantification of infant brains has been widely investigated. According to Li et al. (2019), four distinct approaches for the aforementioned problem are: atlas-based, deformable-surface-based, learning-based, and hybrid. Public and widely available datasets, such as iSeg17 and iSeg19, have boosted the research in the isointense phase. Within the scope of this research, the methods implemented for the segmentation of the isointense brain can be split into three categories. These methods include atlas-based, learning-based (further split into classical machine learning and deep learning based), and hybrid-based approaches, all of which have state-of-the-art representations that can be seen in Table 1.

2.1. Atlas Based Approaches

Atlas-guided segmentation utilizes deformation map between a fixed and moving image to propagate corresponding segmentation labels to unlabeled data (Iglesias and Sabuncu (2015), Li et al. (2019)). This method of segmentation is typically computationally expensive since a non-linear deformation map must be constructed between the non-similar images (Wang et al. (2015)).

In general, creating an atlas requires either an image of an individual or multiple images from different individuals averaged together. Due to high variability in the anatomy between individuals, a multi-atlas label fusion approach is commonly used (Wang et al. (2015))

The features of the MR data determine whether or not T1-weighted or T2-weighted images should be employed for image registration at a specific phase in brain development. Given that an atlas requires adequate contrast between classes, T2-weighted images are typically employed in neonates (Shi et al. (2010a), Shi et al. (2010b)), whereas T1-weighted images are used in adults.

de Macedo Rodrigues et al. (2015) proposed a segmentation pipeline for infants aged 0 to 2 years old (covering all three distinct phases in brain development). In this methodology, a multi-atlas label fusion approach

Methodology	Author	CSF	GM	WM	Dataset
Atlas-Based Methods					
Infant FreeSurfer	de Macedo Rodrigues et al. (2015)	75	77	73	iSeg17
Machine Learning					
LINKS	Wang et al. (2015)	92.6	86.5	87.1	50 infant images
SDM + HF + RF	Wang et al. (2018a)	92.5	90.5	89.4	50 infant images
Deep Learning					
U-Net	Wang et al. (2018b)	92.7	89	91.9	iSeg17
SDM + U-Net	Wang et al. (2018b)	95.8	**92.3**	**93.3**	iSeg17
DenseNet + Skip Connections	Wang, Li, et al. (2019)	**96**	92.1	90.8	iSeg17
Hybrid Methods					
JLF + FE	Wang, Li, et al. (2019)	89	90.3	93.1	iSeg17

Table 1: Overview of the state of the art methods and the respective Dice Similarity coefficient (DSC) score used for segmentation of the isointense phase. The methods presented are comparable with respect to results considering the dataset used for validation
SDM stands for signed distance maps, HF stands for Haar Features, RF stands for Random Forest, JLP stands for Joint Label Fusion and FE stands for Feature Extraction
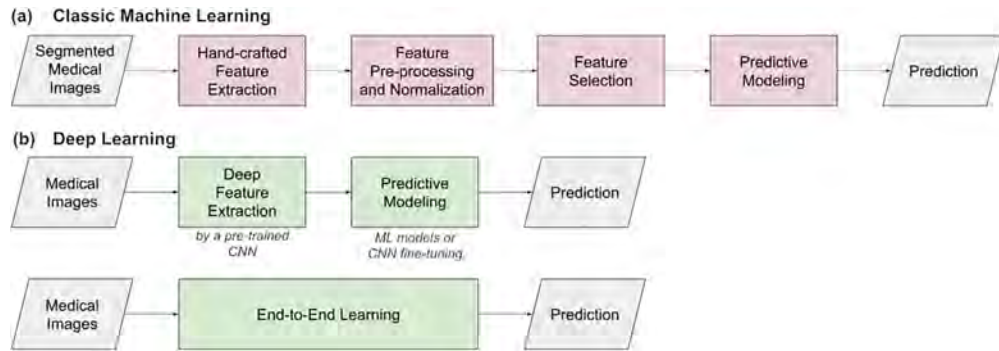

Figure 2: Visual representation of the typical learning based approach workflow. Courtesy of Castiglioni et al. (2021)

was used. However, the performance in the isointense phases is worse than in the neonatal period, probably a result of the weak contrast in both T1-weighted and T2-weighted images during the isointense period.

2.2. Learning Based Approaches

Machine learning and deep learning are two types of learning-based techniques commonly used in medical imaging (Castiglioni et al. (2021)). Figure 2 depicts a typical workflow for the aforementioned approaches.

In general, machine learning approaches require explicit data characterization through the extraction of hand-crafted features, whereas deep learning models learn more representative and discriminative features automatically (Moeskops et al. (2016), Li et al. (2019),Castiglioni et al. (2021)).

Due to the public availability of iSeg17 and iSeg19 datasets, many learning-based approaches have been developed to work with isointense MR data.

For instance, in a sequential random forest approach for segmentation, the LINKS technique developed by Wang et al. (2015) includes multi-modal Haar-like feature information and previous tissue probability maps prediction Haar-like features. One of the teams from the iSeg17 challenge created a hybrid model based on

the implementation of features of both original T1-weighted and T2-weighted images as well as features from obtained joint tissue probability maps (Sanroma et al. (2016)). The features above were fed into SVM for classification.

Bui et al. (2017) and Dolz et al. (2020) implemented a network based on DenseNet. The former introduces skip connections between respective layers to increase the amount of contextual information captured, whereas the latter combines information from all the previous convolutional layers in the final block of the network to prevent resolution loss and preserve gradient flow.

Lei et al. (2019) introduced a U-Net with an attention mechanism to approach the problem of blurred tissue boundary between WM and GM.

As stated in Wang et al. (2019) and Sun et al. (2021), the majority of the methods rely on the application of a well-established method or an advanced deep learning method. However, they do not introduce any information that can be used to guide segmentation and which can be determined via qualitative MRI analysis.

Not leveraging the prior anatomical knowledge, e.g., cortical thickness is within a certain range, WM is a topological sphere enveloped by GM (Huang and

Roberts (2021)) leads to misalignments (holes and handles) in the border between WM and GM in the cortical regions of the brain as can be seen from Figure 3.
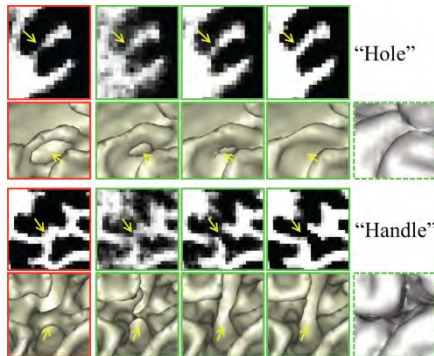

Figure 3: Visual representation of the holes and handles present in non-anatomically guided tissue segmentation. Figure was adapted from fig. 7 in Wang et al. (2018a)

2.2.1. Introduction of Anatomical Knowledge

Machine Learning

The aforementioned limitation was approached in Wang et al. (2018a) by utilizing signed distance maps. A signed distance map gives information on the closest distance between any point and the border. By introducing the signed distance map, the anatomical constraint of GM enveloping WM has been approximately satisfied (Huang and Roberts (2021)). A two-stage sequential random forest approach was devised to obtain anatomical information by first segmenting CSF against WM and GM, taking into account that CSF has a higher contrast towards brain tissue than WM and GM have between each other. This was followed by using the combined segmentation of WM and GM to generate a signed distance map to be utilized as a channel for the next phase in segmentation (Figure 4). Similarly, as before, sequential random forest and Haar-like features were used to obtain final segmentation results.
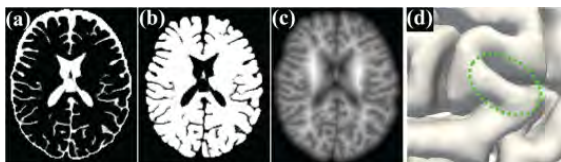

Figure 4: The results of the estimated (a) CSF, (b) GM and WM combined and (c) obtained signed distance map with respect to the outer surface (as shown in (d)) from the GM and WM combined segmentation. Courtesy of Wang et al. (2018a)

The promising effects of introducing anatomical knowledge can be seen from quantitative results showcased in Table 1. An increase in DSC score has been observed with respect to both GM and WM when the algorithm were tested on the same validation dataset.

Deep Learning

Wang et al. (2018b) implemented a similar methodology as previously, but using a deep learning algorithm consisting of a U-Net with DenseBlocks in encoder and decoder parts. A first network was trained for a binary classification between CSF and the combination of WM and GM. The obtained WM and GM segmentation was utilized to derive a signed distance map, which was used as an additional channel for the following network architecture for the segmentation of WM and GM.

Table 1 showcases an improvement in segmentation of both GM and WM in comparison to the other implementation. Moreover, an increase of 2.5% in DSC score was achieved in segmenting WM in comparison to the best result of iSeg17 challenge (Wang et al. (2019)). Furthermore, it can be observed that, in general, deep learning methods lead to better results than other methods.

2.3. Limitation of the Current Methods

As already stated, many deep learning algorithms were used to approach the challenging isointense phase of brain development (Zhang et al. (2015), Nie et al. (2016), Wang et al. (2019), Sun et al. (2021)). However, only a few take into account prior anatomical information that can guide the segmentation process. Furthermore, limited pre-processing is applied. Moreover, while approaches achieve promising results, errors on the GM/WM border remain.

Signed Distance Maps (SDM) are one of the strategies that can be utilized as prior knowledge to constrain segmentation and produce good results both qualitatively and quantitatively (Wang et al. (2018b)). However, the method requires training two separate models to generate the respective SDM. Considering there are other ways to introduce anatomical constraints (through shape, appearance, motion, and context information (Liu et al. (2021))) while only training a single model, this is time inefficient and introduces more complexity.

2.4. Contributions of this work

Inspired by the work of Wang et al. (2018b) and Wang et al. (2018a), a new approach to dealing with the isointense stage of brain development is developed. This approach manipulates previously defined tissue labels on reference MR images (atlas) as prior knowledge to segment a target image. Mainly, a WM prior is employed as the neural network's third input channel, combined with T1-weighted and T2-weighted images.

Furthermore, considering the numerous ways of introducing prior information and prevailing error present on a GM/WM border, a multi-branch network was developed motivated by the works of Navarro et al. (2019). Contour prediction is added as the network's output channel keeping in mind the network's bias towards texture rather than shape.

23.4

3. Material and methods

3.1. Dataset and atlas

3.1.1. iSeg19

The dataset used in this paper is the publicly available dataset provided by the iSeg19 challenge (Sun et al. (2021)). This dataset consists of T1-weighted and T2-weighted images of infants aged 6±0.8 months. Ten training samples containing both intensity and ground truth (GT) images, thirteen cases of only intensity images for validation, and sixteen cases for testing were acquired in multiple centers.

According to the organizers of the challenge, annotation of CSF, WM, and GM of the training dataset were obtained using a longitudinal guided segmentation algorithm (Wang et al. (2013)). The later time point volume of a subject is used to segment the earlier time point volume. Specifically, later time point segmentation is utilized to guide robust segmentation of the neonate (which is based on convex optimization and coupled level sets) through guided level sets. This segmentation was followed by expert manual refinement of the results. However, the initial segmentation of the validation dataset was done using anatomy-guided densely connected U-Net (Wang et al. (2018b)). In addition, the following steps were taken to unify the dataset: (i) all images were resampled to 1x1x1 mm^3, (ii) skull stripping, (iii) intensity inhomogeneity correction, and (iv) removal of the cerebellum and brain stem.

Through visual analysis of the subjects's raw images, as well as the corresponding image intensity histograms, the following conclusion was drawn about the dataset: (i) both T1-weighted and T2-weighted images display highly overlapping intensities between WM and GM; (ii) unlike T2-weighted images, T1-weighted images display a distinct peak for the CSF class; (iii) T2-weighted images display a CSF with evenly distributed intensities; (iv) in T1-weighted images, the subjects 4 and 7 have different higher intensities than the other cases.

3.1.2. Atlas

An atlas (and its associated segmentation) is one of the possible ways of imposing prior information. The atlas should provide information on common brain anatomy and ensure that the WM is a topological sphere.

Zhang et al. (2016) provide a 6 months old infant atlas of common brain anatomy in a standardized space (Figure 5). Specifically, this atlas is obtained by deconstructing longitudinally collected MRI volumes using wavelets and then accommodating spatial and temporal variability into these volumes using group-sparse construction.
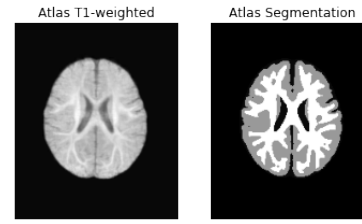
Figure 5: Atlas of a 6-month old infant and its segmentation provided by Zhang et al. (2016).

3.2. Data Preparation

Pre-processing the data reduces the variability across subjects. Firstly, in each image, the intensities above the 99 percentile and below 1 percentile were cropped, removing the influence of outliers. Secondly, a min-max normalization was used to scale the values between 0 and 1. Finally, a data augmentation technique of flipping was implemented to increase the number of volumes during training. The dataset was doubled by including left-to-right flipped volumes in training, taking into account the pseudo-symmetrical nature of the left and right hemispheres of the brain. The described pre-processing and data augmentation is depicted in Figure 6.
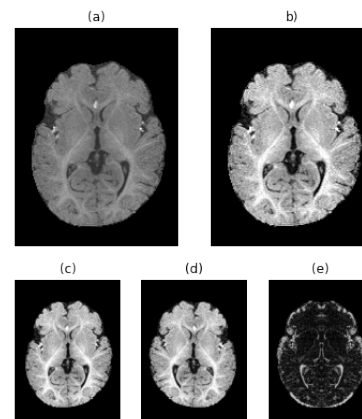
Figure 6: Visual representation of data pre-processing and augmentation steps: (a) axial slice of the original T1-weighted, (b) processed image with quantile min-max; (c), (d) illustration of an original and flipped and (e) illustration of the absolute difference between the original and flipped slice to indicate the difference between the volumes.

3.2.1. Anatomical Prior

Image Registration

To derive the anatomical prior, image registration was implemented to align the atlas and the particular subject. Specifically, the T1-weighted image of the subject and the T1-weighted atlas were registered.

Two different strategies were employed for image registration, affine, and affine and non-rigid registration combined.
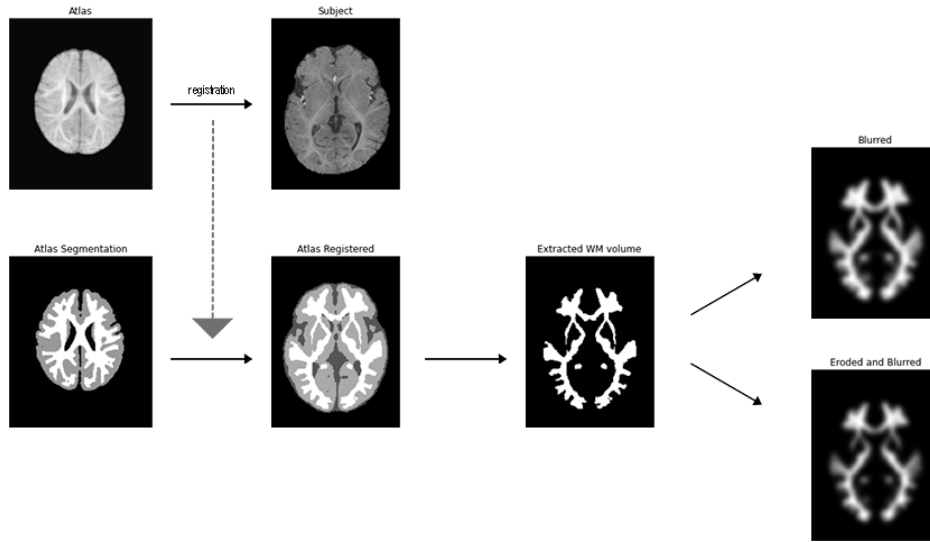
Figure 7: Pipeline for extracting prior information from the atlas

The affine registration implemented is based on the Aladin algorithm (Ourselin et al. (2001)). It is a two-step coarser to finer detail scheme registration. A collection of corresponding points between reference and target is produced via block-matching and filtered via cross-correlation. The affine registration results were improved by subsequently applying non-rigid registration. In particular, a B-spline model implemented by Modat et al. (2010) was used. The implementation consists of three main steps (i) modification of the floating image using splines and interpolation, (ii) evaluation of objective function, and (iii) optimization of the function.

Due to the time complexity of optimizing image registration, default parameters were utilized for affine and non-rigid registrations. For affine, they include three levels of the coarser-to-finer scheme with a maximum of 5 iterations per level before reaching the next level. The non-rigid registration was provided with initial affine transformation. The final grid spacing for the spline on the x,y, and z-axis was set to 5 voxels. The number of bins for calculation of similarity measure was set to 64.

Prior Information Processing

Registration of the T1-weighted atlas to the subject's T1-weighted image was followed by propagating atlas labels to the subject space. The propagation of the labels was followed by extracting only the WM segmentation and processing via two distinct strategies. The first strategy included only applying Gaussian blurring and the second included applying erosion and then Gaussian blurring, as can be seen in Figure 7.

The Gaussian blurring was implemented to smooth the anatomical prior, in order to make the binary WM prior look more similar to the signed distance map (Figure 4). The blurring level was adjusted and sigma was set equal to 3 as this choice preserved reasonable information from the prior.

Given the imperfect alignment between the ground truth and the propagated labels, the practice of applying erosion and then Gaussian blurring was used to provide more certain information on the presence of WM at specific locations.

3.2.2. Patch Extraction

As a result of GPU memory constraints, a patch extraction tactic was employed.

The patch extraction was done considering different patch sizes and patch strides. The overlap was introduced by decreasing the patch stride to be smaller than the patch size. Furthermore, considering that a significant component of the subject's MR volume is empty space around the brain, only patches that include at least a predefined portion of genuine brain volume are considered in training.

To find the ideal network settings, the following combinations of four patch sizes and patch strides were taken into account:

1. 16x16x16 and 16x16x16
2. 16x16x16 and 8x8x8
3. 32x32x32 and 32x32x32
4. 32x32x32 and 16x16x16

3.3. Deep Learning Network

U-Net is a well-known state-of-the-art deep learning network for biomedical image segmentation introduced by Ronneberger et al. (2015). This network has been extensively used to segment the isointense phase of brain development (Sun et al. (2021); Wang et al. (2018b)). Consequently, a 3-D U-Net architecture is used in this project.
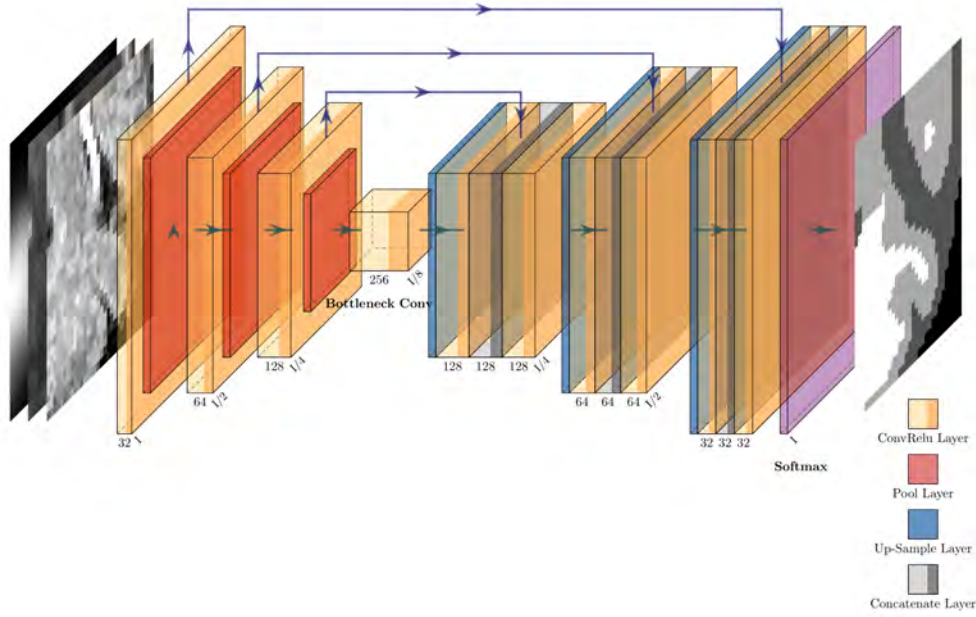
Figure 8: U-Net model used for segmentation.*Disclaimer: The figure illustrates just a slice but the actual model uses a 3D patch.*

The U-Net architecture consists of two symmetric parts: encoder (down-sampling) and decoder (up-sampling). The encoder extracts high-order abstract features from images while shrinking their size, whereas the decoder gradually restores the input image's original size. Furthermore, skip connections between the encoder and decoder preserve information.

A sequential process of repeated convolutions and ReLU activation functions is utilized to extract high-order abstract features, followed by max-pooling procedures. The number of features doubles with each max-pooling action. The decoding counterpart is reversed. The feature maps are concatenated with their down-sampling complements before passing through convolution and ReLU, followed by up-sampling. After each up-sampling, the number of features is decreased by two. The soft-max activation function and 1x1x1 convolution are the network's last steps to provide the per-class probability at each voxel.

Furthermore, to prevent overfitting, dropout is introduced. In the encoder, it is applied before max-pooling and in the decoder before up-sampling. Each of the three encoding blocks in the model included a single 3D convolution, ReLU activation, and dropout before max-pooling. In the decoder, up-sampling was followed by concatenation with the matching encoder counterpart before performing a single 3D convolution and dropout.

Considering the main topic is observing the effects of introducing prior information to the network architecture, a simple U-Net model was employed as seen from Figure 8 using parameters displayed in Table 2.

Training Parameters	
Optimizer	Adam
Loss	Categorical Cross-Entropy
Batch Size	32
# of Epochs	100
Patience	10

Table 2: Overview of training parameters used for training the U-Net

3.4. Multi-branch U-Net

A complementary-task learning U-Net was developed to deal with the issue of tissue segmentation with the presumption that the contour prediction will boost the performance on the border between WM and GM. The tissue contrast between CSF and GM is much higher than the tissue contrast between WM and GM, and as a result, the network was created to only predict the WM/GM contour. Moreover, the model was trained only on T1-weighted and T2-weighted images for the multi-branch network.
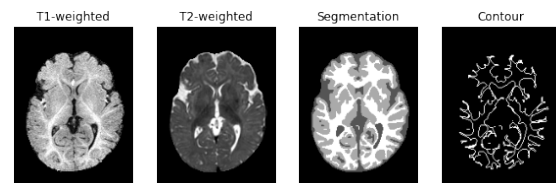

Figure 9: Corresponding T1-weighted and T2-weighted images of Subject 1, and target segmentation and contour map (binary edge of GM/WM border)

The U-Net architecture, in this case, consisted of

common encoding layers and common decoding layers until the final one. The second to last decoder block splits into two separate decoder blocks, each with its own softmax activation function. The segmentation is output by one, while the contour is output by the other. As there are two separate outputs, the loss function optimized objectives of related tasks, segmentation (categorical cross-entropy) and contour prediction (binary cross-entropy).

$$L_{total} = L_{tissue} + L_{contour}$$

$$L_{tissue} = -\sum_{c=1}^{M}\sum_{o=1}^{M} y_{o,c} \log(p_{o,c})$$

$$L_{contour} = -\sum_{c=1}^{M}(y_c \log(p) + (1 - y_c)\log(1 - p))$$

where M - number of classes, y - binary indicator (0 or 1) if class label c is the correct classification for observation o, and p - predicted probability that observation o is of class c.

3.5. Training and Validation Strategy

As a result of the limited number of subjects available for the training and validation, a leave-one-out cross-validation was employed during training to obtain the optimal parameters and evaluate the effects of introducing prior information. During network training, an 80%/20% strategy was employed, the former being the % of training subjects and the latter being the validation subjects. One case would be left as the test sample to observe the final results.

Firstly, the effects of introducing a prior obtained using affine label propagation to the network as the third channel to the input were observed. The initial patch size and stride were both set to 16x16x16. Secondly, the introduction of flipped counterparts (which double the dataset) was observed. Thirdly, the best parameters for patch and patch stride were identified. Once the optimal parameters have been set, the effect of the prior was scrutinized by:

- Training the network using the improved prior. The improved prior was firstly obtained by employing affine and non-rigid label propagation. Furthermore, pre-processing the prior also included the utilization of erosion and then Gaussian blurring.

- Training the network with the same parameters without prior information

- Predicting the segmentation *(of the network trained with WM prior)* using GM prior

- Training the network using a "perfect" prior obtained using the available GT to observe the possible results of the best-case scenario concerning image registration and acquired prior. The "perfect" prior was obtained by extracting the WM volume from the ground truth, eroding it, and blurring it before feeding it to the network

In addition, a multi-branch U-Net with optimized parameters was evaluated using auxiliary information regarding the border of GM and WM.

The best-performing pipeline was further tested on the iSeg19's validation dataset. Due to the low number of training, it was trained on the entire training dataset. In addition, because GT is unavailable for the validation dataset, the challenge organizers conduct the validation independently after receiving your segmentation results.

3.6. Quantitative Analysis

To evaluate the accuracy of the proposed segmentation, Dice Similarity coefficient (DSC) and Volumetric Difference were computed.

3.6.1. Dice Similarity Coefficient

Denoting GT ground truth and S as segmentation of each MR volume, the DSC was defined by the following formula:

$$DSC = \frac{2|GT| \cap |S|}{|GT| + |S|}$$

DSC values range between 0 and 1 corresponding to the worst and best overlap between the ground truth and prediction.

3.6.2. Volumetric Information

For the volumetric information, the difference between the GT segmentation and the model segmentation was observed concerning:

1. Total intracranial volume (WM + GM + CSF volumes combined)
2. GM volume
3. WM volume

3.7. Implementation Details

This project was implemented using Python programming language. Complementary libraries used include numpy, nibabel, patchify and matplotlib. Image registration was done using icometrix's implementation of NiftyReg. Erosion and blurring of the registered atlas were done using scikit-image package (van der Walt et al. (2014)). The 3D U-Net was implemented using Tensorflow (Abadi et al. (2015)). Furthermore, the visualization was done in 3D Slicer (Kikinis et al. (2014)).

4. Results

Following the description in subsection 3.5. *Training and Validation Strategy*, the results section will be split to cover the indicated experiments.

4.1. Introducing Prior Information

The first experiment evaluates if the addition of prior information on a simple network with no optimized parameters could led to improved segmentation results. Table 3 shows the result of this experiment, comparing a baseline segmentation that uses simple label propagation from atlas to subject image against results obtained with U-Net models. Cross validation results of a U-Net without prior information, with WM guidance and with WM guidance and data augmentation (flipping of the axis) are included in the table. Notice that the performance increases when using a 3D U-Net rather that label propagation, while a slight increase in the WM performance is observed when adding prior information guidance of this tissue.

As data augmentation produces a slight increase in performance for all the 3 classes, the flipping of the axis is kept during training in the subsequent steps.

	CSF	GM	WM
	Baseline		
Affine label propagation	0.53 (0.0180)	0.66 (0.0159)	0.62 (0.0082)
	U-Net		
No Prior	0.886 (0.0158)	0.833 (0.0134)	0.787 (0.0183)
With WM Prior	0.886 (0.0158)	0.839 (0.0110)	0.801 (0.0110)
WM Prior, Data Augm	0.890 (0.013)	0.840 (0.006)	0.808 (0.019)

Table 3: Average DSC and corresponding std. deviation of segmentations when only using affine registration. Furthermore, a comparison of DSC results when not using prior, with WM prior, as well as combined WM prior and introduction of data augmentation (flipped axis).

4.2. Patch size and stride

The introduction of overlapping patches led to an increase in the number of sample during training. When using a patch size of 16, the number of patches increased from around 1300 to around 21000. On the other hand, in the case of patch size of 32, the number increased from around 159 to around 4000. The exact number of patches varies between training in the leave-one-out strategy, considering that different sub-set of subjects is used to extract patches during each training.

Table 4 shows the DSC when changing the patch sizes and strides to include overlapping patches. The best performance score is obtained when using the patch size of 16 and stride of 8. Hence, the next steps will be done using those parameters.

		CSF	GM	WM
Patch Size	*Patch Stride*			
16x16x16	*16x16x16*	0.89 (0.1310)	0.84 (0.00650)	0.808 (0.01869)
	8x8x8	**0.919 (0.0111)**	**0.879 (0.0083)**	**0.854 (0.0159)**
32x32x32	*32x32x32*	0.88 (0.0186)	0.828 (0.0156)	0.791 (0.0242)
	16x16x16	0.918 (0.0115)	0.874 (0.0099)	0.844 (0.0153)

Table 4: Average DSC and corresponding std. deviation of segmentation results when using different patch sizes and strides.

4.3. Utilization of Non-Rigid Registration to obtain prior

The next experiment improved the WM prior by sequentially combining affine and non-rigid registration and propagating corresponding labels. The overall improvement in the prior can be seen by comparing the DSC between only affine registration (see Table 3) and combined (see Table 5).

	CSF	GM	WM
Affine+nonrigid label propagation	0.679 (0.0098)	0.717 (0.0112)	0.713 (0.0159)

Table 5: DSC and and corresponding std. deviation of segmentations obtained using combined affine and non-rigid label propagation.

4.4. Improved Prior

With the refinement of prior through the usage of combined affine and non-rigid label propagation, no improvement in the DSC can be observed (compare Table 4 (patch size 16x16x16 and stride 8x8x8) and Table 6 (With WM Prior)) when training the knowledge-guided U-Net for segmentation.

	CSF	GM	WM
	WM Trained U-Net		
With WM Prior	0.919 (0.0119)	0.879 (0.0096)	0.854 (0.0157)
With WM Prior (E + GB)	**0.92 (0.0106)**	**0.88 (0.0088)**	**0.855 (0.0156)**

Table 6: Average DSC and std. deviation of segmentation when using optimized patch size and patch stride. *"With WM Prior (E+GB)" stands for a model trained using the prior obtained with the pre-processing including combined erosion and Gaussian blurring.*

With the idea to overcome the presence of considerably imprecise information in cortical regions, the eroded prior should offer a more precise prior of the WM localization. Hence, the following test in improving the prior also considers the application of erosion before Gaussian blurring to the prior. Overall, an increase of 0.1% is visible for both CSF and WM classes when using the combined pre-processing (see Table 6).

Besides the evaluation of the DSC, by analyzing the volumetric information using the prior obtained with combined pre-processing, it can be concluded that the absolute difference in volumes between segmentation and GT for both GM and WM leads to very similar average values. By observing the non-absolute difference on a case-by-case basis, it can be inferred that in most cases, segmentation using the proposed pipeline leads
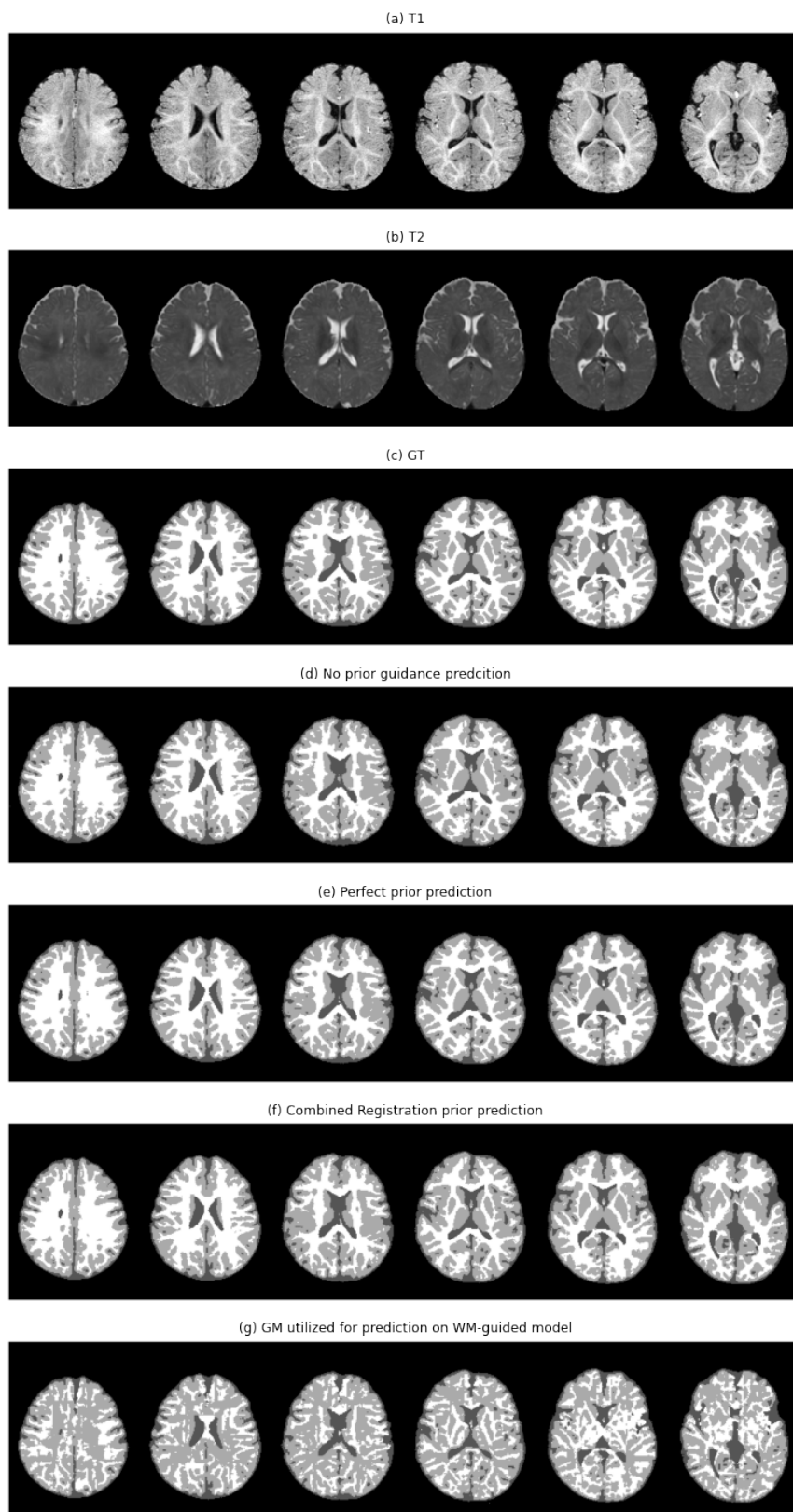
Figure 10: Visual illustration of Subject 1's (a) T1 images, (b) T2 images, (c) GT, (d) segmentation results when using no WM guidance, (e) segmentation results when using the perfect prior, (f) segmentation results when using prior obtained through combined registration, and (g) segmentation results when using GM prior for prediction on a WM prior trained network.
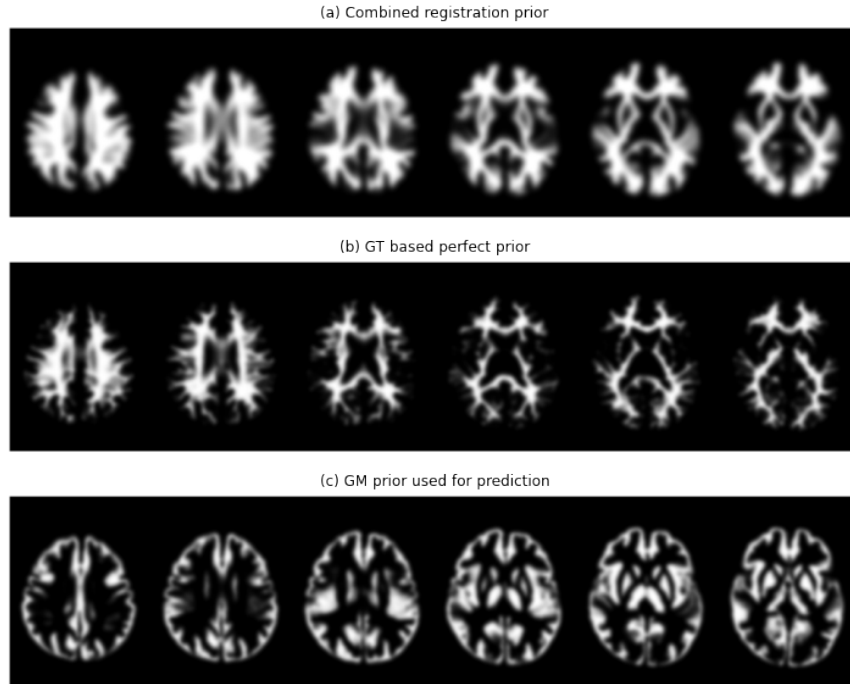
Figure 11: Visual illustration of Subject 1's (a) WM prior obtained using combined registration; (b) WM prior obtained using GT, and (c) GM prior used for prediction.

Subject	diff [ml]	diff CSF [ml]	diff GM [ml]	diff WM [ml]
1	-0.559	3.537	25.088	-29.184
2	-1.02	-1.902	3.711	-2.829
3	-1.269	-3.78	33.43	-30.919
4	-0.467	3.111	26.257	-29.835
5	-1.034	-9.657	31.168	-22.545
6	-0.89	-13.685	14.989	-2.194
7	-2.546	5.093	-4.974	-2.665
8	-0.716	0.606	9.914	-11.236
9	-1.083	7.326	35.908	-44.317
10	-1.508	-7.695	-41.217	47.404
Abs average	1.1092	5.6392	22.6656	22.3128

Table 7: The 1st column shows difference in total intracranial volume between segmentation and GT, 2nd-4th column show difference in the volume between CSF, GM, and WM respectively between segmentation and GT. The last row display the average absolute difference between corresponding values.

	CSF	GM	WM
No WM Prior	0.918 (0.0108)	0.879 (0.0071)	0.853 (0.0151)

Table 8: Average DSC and std. deviation of segmentation when using optimized patch size and patch stride with no prior as the third channel

visual analysis, the proposed pipeline (with using the prior) leads to more refined results. The improvement in using the prior guidance is noticeable on the cotical and sub-cortical border between GM and WM (see Figure 12).
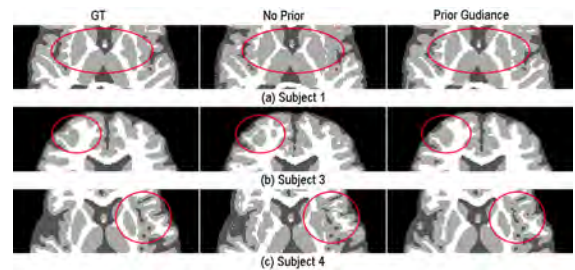
Figure 12: Improvement in segmentation when using the prior. The 1st column shows the GT of corresponding subject, 2nd column showcases the segmentation results when not using the WM prior and 3rd column when using the prior obtained using combined registration and pre-processed with erosion and Gaussian blurring.

to over-segmentation of GM and under-segmentation of WM (see Table 7).

4.5. No Prior

In order to see the effects of the prior, a network with previously optimized parameters (patch stride and size) was trained without the input prior as the third channel.

Overall, using no prior information for training leads to similar quantitative results as when using the prior (see Table 8). The results are slightly worse by a difference in DSC of 0.2%, 0.1%, and 0.2% for each of the CSF, GM, and WM classes, respectively, when compared to using prior obtained by erosion and Gaussian blurring (see Table 6).

Although there is a slight difference in DSC, through

4.6. Utilization of GM for prediction

The results of segmentation with and without the prior display similar DSC (see Table 6 and Table 10).

In order to test if the network utilized the prior for prediction, GM prior was fed into the network trained on WM prior.

GM prior was obtained in a similar manner as WM prior.

	CSF	GM	WM
WM Trained U-Net			
GM prior	0.886	0.754	0.604

Table 9: DSC of the segmentation obtained using GM prior for prediction with the network trained with WM prior.

DSC results from Table 9 showcase that the network predicts poorly when misleading information is given concerning the prior. The DSC decreased by 3.4%, 12.6%, and 25.1% for CSF, GM, and WM, respectively.

Visual results from using the GM prior to predict can be observed in Figure 10.(g). By reviewing the Figure, it is clear that using GM prior on a WM prior trained network misleads the prediction. It leads to almost reversed GM and WM in prediction. Hence, the experiment conclusively shows that the network utilizes the prior for the prediction.

4.7. "Perfect" Prior

The proposed prior information is not optimal as a consequence of the registration errors between the atlas and subjects, mainly due to the lack of intensity contrast and anatomical heterogeneity. Hence, the full potential of adding prior is not fully visible. Aiming to study the effects of this type of guidance in the best-case scenario of a "perfect" prior (see Figure 7, (b)) was utilized. The "perfect" prior gives much-refined information in the cortical regions of the brain, which is different from the prior obtained using combined label propagation with much fuzzier cortical region information.

	CSF	GM	WM
With Perfect Prior	0.929 (0.0085)	0.925 (0.0047)	0.927 (0.0077)

Table 10: Average DSC of segmentation when using optimized patch size and patch stride with *perfect* prior

Using the "perfect" prior increases the DSC for all 3 of the classes (Table 10). 0.9% in case of CSF, 4.5% in case of GM, and 7.2% in case of WM when compared against the results of the model with best results (Table 6).

This experiment concludes that improvement in the registration could lead to improved results in the final segmentation.

4.8. Complementary task learning

As a result of misalignments on the GM/WM border, the next step included analyzing if introducing complementary task learning can aid the segmentation of isoin-

tense brain tissue. In this case, only the T1-weighted and T2-weighted images were used for training.

	CSF	GM	WM
Segm + Contour pred	0.915 (0.0116)	0.874 (0.0094)	0.85 (0.0146)

Table 11: Average DSC and std. deviation of segmentation predicted using complementary task learning.

Adding the contour as a complementary task to learn does not improve the results. Specifically, the decrease in DSC of all three classes is observed compared to the network trained using the WM prior as the input, as well as the network trained without using the WM prior (see Table 11).

4.9. iSeg19 Validation Dataset

The final model included using a U-Net with patch size and stride of 16 and 8, respectively, T1-weighted, T2-weighted, and prior obtained using affine and non-rigid label propagation processed with erosion and Gaussian blurring as input.

The number of epochs used for training was 42. That number was obtained by averaging the number of epochs required for convergence during the leave-one-out strategy.

As mentioned earlier, GT is unavailable for the validation dataset; hence, the challenge organizers conduct the validation independently. Therefore, all the results showcased are received from the challenge's organizers.

	DICE	HD	ASD
CSF			
Mean	0.927	10.023	0.197
Std	0.007	1.659	0.017
Min	0.912	7.681	0.174
Max	0.939	11.57	0.225
GM			
Mean	0.891	7.340	0.437
Std	0.009	1.247	0.038
Min	0.873	5.657	0.390
Max	0.904	9.434	0.541
WM			
Mean	0.860	6.659	0.499
Std	0.016	1.020	0.048
Min	0.824	5.000	0.453
Max	0.874	8.775	0.622

Table 12: Average mean, std. deviation, min and max values for DSC, HD (95-th percentile Hausdorff Distance), and ASD (average surface distance on the validation dataset of the iSeg19 challenge.

As a result of proper training (e.g., avoiding overfitting) and consequently a good generalization performance, the segmentation results are coherent with what was obtained during training. Observing the difference between the mean DSC of training (Table 6, *With WM*
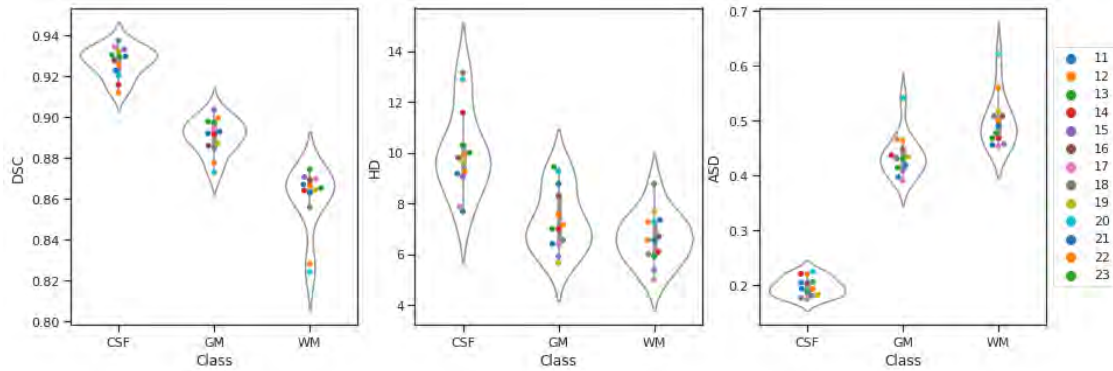
Figure 13: Performance of the pipeline on the validation dataset of iSeg19 challenge in terms of DSC, HD (95-th percentile Hausdorff Distance), and ASD (average surface distance), using violin plots. Each color from the legend represents the Subject number from the dataset.

Prior (E + GB)) and validation (Table 12), a slightly higher DSC values in segmentation results of all 3 classes is seen. 0.8% in case of CSF, 1.1% for GM, and 0.5% for WM.

Figure 13 illustrates values for each one of the metrics used by the organizers of the challenge, DSC, HD, and ASD, on a case-by-case basis. It can be observed that when segmenting WM and GM worst results are obtained for Subject 20 (DSC of 0.824 and 0.873 for the classes mentioned above, respectively). That can be due to Subject 20 being slightly rotated (see Figure 14) compared to other cases leading to erroneous segmentation.
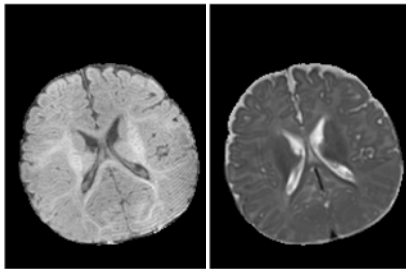

Figure 14: Axial slice of T1-weighted and T2-weighted validation subject number 20. Notice the severe rotation.

5. Discussion and Concusions

The presented model deals with the segmentation of infant brain tissue in the isointense phase, characterized by the limited distinction between GM and WM tissue on T1-weighted and T2-weighted images. The limited contrast leads to difficulties even in manual delineation.

The segmentation was performed using knowledge-guided segmentation by considering the potential of using anatomical prior to guiding segmentation and the limitations presented in the previous work. Primarily, tissue segmentation was formulated as a patch segmentation task where the relationship between patches was

not considered. Furthermore, only patches that contained a pre-defined amount of brain anatomy were utilized. Secondly, prior guidance was introduced in a two-fold way. The foremost way included obtaining an anatomical prior to guiding the segmentation by taking advantage of the available atlas for the 6-month-old infant. In particular, combined affine and non-rigid image registration were utilized to register the atlas to the patient space, and subsequently, WM was extracted and processed using erosion and blurring. Once the prior was obtained, segmentation was achieved by employing a U-Net with multiple inputs, considering T1-weighted, T2-weighted images, and the derived WM anatomical prior. The second way included introducing the prior by developing a complementary task learning through multi-output U-Net segmentation of the tissue classes and GM/WM border. In this case, the neural network would utilize the WM/GM border prior as an output to penalize the segmentation.

Overall, experimental results on the iSeg19 dataset showcase a slight increase in the quantitative results when utilizing the anatomical prior as the third input channel, obtained with combined image registration and processed with erosion and Gaussian blurring. Visual analysis of the above results demonstrates more refined segmentation in sub-cortical regions when using the prior. The resulting slight quantitative difference between the models can be due to the atlas, in general, being blurry in cortical regions of the brain and consequently not providing precise information.

5.1. Limitations

Despite the good and coherent results of the model, some limitations can be improved with further work. The model struggles to segment the region on the border between WM and GM, usually leading to the over-segmentation of GM and under-segmentation of WM. This results from a lack of intensity contrast in this area as myelination processes from central to peripheral brain regions. Furthermore, when unusual poses (rotations) are present, erroneous segmentation results are

achieved. In addition, very few data samples are available for training the network.

When it comes to complementary task learning, in some cases, the segmentation loss will drop below the segmentation loss obtained without complementary task learning. The final segmentation produced was worse because neither the choice of the loss function nor the weighting between the two losses was optimized.

5.2. Future Work

In order to improve the segmentation results, further optimizing the image registration between the atlas and the subject could add more precise information in cortical regions of the brain. In addition, considering that the most erroneous region is the border between WM and GM, increasing the number of patches selected from that region can potentially improve the segmentation results. Introducing data augmentation with more aggressive rotations could lead to a more robust method.

Further network optimization concerning the number of encoding/decoding blocks, the number of convolutional layers, and the introduction of batch normalization could potentially improve the results.

Finally, the usage of more data points would improve the network's overall performance, considering that the performance of the DL network depends on the amount of available data.

Acknowledgments

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. software available from tensorflow.org.

Adam, A., Dixon, A.K., Gillard, J., Schaefer-Prokop, C., Grainger, R.G., Allison, D.J., 2014. Grainger & Allison's Diagnostic Radiology E-Book. Elsevier Health Sciences.

Bui, T.D., Shin, J., Moon, T., 2017. 3d densely convolutional networks for volumetric segmentation. arXiv preprint arXiv:1709.03199 .

Castiglioni, I., Rundo, L., Codari, M., Di Leo, G., Salvatore, C., Interlenghi, M., Gallivanone, F., Cozzi, A., D'Amico, N.C., Sardanelli, F., 2021. Ai applications to medical images: From machine learning to deep learning. Physica Medica 83, 9–24.

Damiano, C.R., Mazefsky, C.A., White, S.W., Dichter, G.S., 2014. Future directions for research in autism spectrum disorders. Journal of Clinical Child & Adolescent Psychology 43, 828–843.

Dolz, J., Desrosiers, C., Wang, L., Yuan, J., Shen, D., Ayed, I.B., 2020. Deep cnn ensembles and suggestive annotations for infant brain mri segmentation. Computerized Medical Imaging and Graphics 79, 101660.

Elder, J.H., Kreider, C.M., Brasher, S.N., Ansell, M., 2017. Clinical impact of early diagnosis of autism on the prognosis and parent–child relationships. Psychology research and behavior management .

Gilmore, J.H., Kang, C., Evans, D.D., Wolfe, H.M., Smith, J.K., Lieberman, J.A., Lin, W., Hamer, R.M., Styner, M., Gerig, G., 2010. Prenatal and neonatal brain structure and white matter maturation in children at high risk for schizophrenia. American Journal of Psychiatry 167, 1083–1091.

Haefner, H., Maurer, K., 2006. Early detection of schizophrenia: current evidence and future perspectives. World Psychiatry 5, 130.

Hazlett, H.C., Poe, M., Gerig, G., Smith, R.G., Provenzale, J., Ross, A., Gilmore, J., Piven, J., 2005. Magnetic resonance imaging and head circumference study of brain size in autism: birth through age 2 years. Archives of general psychiatry 62, 1366–1376.

Howell, B.R., Styner, M.A., Gao, W., Yap, P.T., Wang, L., Baluyot, K., Yacoub, E., Chen, G., Potts, T., Salzwedel, A., et al., 2019. The unc/umn baby connectome project (bcp): An overview of the study design and protocol development. NeuroImage 185, 891–905.

Huang, H., Roberts, T., 2021. Handbook of Pediatric Brain Imaging: Methods and Applications. volume 2. Elsevier.

Iglesias, J.E., Sabuncu, M.R., 2015. Multi-atlas segmentation of biomedical images: a survey. Medical image analysis 24, 205–219.

Kikinis, R., Pieper, S.D., Vosburgh, K.G., 2014. 3d slicer: a platform for subject-specific image analysis, visualization, and clinical support, in: Intraoperative imaging and image-guided therapy. Springer, pp. 277–289.

Knickmeyer, R.C., Gouttard, S., Kang, C., Evans, D., Wilber, K., Smith, J.K., Hamer, R.M., Lin, W., Gerig, G., Gilmore, J.H., 2008. A structural mri study of human brain development from birth to 2 years. Journal of neuroscience 28, 12176–12182.

Lei, Z., Qi, L., Wei, Y., Zhou, Y., 2019. Infant brain mri segmentation with dilated convolution pyramid downsampling and self-attention. arXiv preprint arXiv:1912.12570 .

Li, G., Wang, L., Yap, P.T., Wang, F., Wu, Z., Meng, Y., Dong, P., Kim, J., Shi, F., Rekik, I., et al., 2019. Computational neuroanatomy of baby brains: A review. NeuroImage 185, 906–925.

Liu, L., Wolterink, J.M., Brune, C., Veldhuis, R.N., 2021. Anatomy-aided deep learning for medical image segmentation: a review. Physics in Medicine & Biology .

Lord, C., Elsabbagh, M., Baird, G., Veenstra-Vanderweele, J., 2018. Autism spectrum disorder. The lancet 392, 508–520.

de Macedo Rodrigues, K., Ben-Avi, E., Sliva, D.D., Choe, M.s., Drottar, M., Wang, R., Fischl, B., Grant, P.E., Zöllei, L., 2015. A freesurfer-compliant consistent manual segmentation of infant brains spanning the 0–2 year age range. Frontiers in human neuroscience 9, 21.

Modat, M., Ridgway, G.R., Taylor, Z.A., Lehmann, M., Barnes, J., Hawkes, D.J., Fox, N.C., Ourselin, S., 2010. Fast free-form deformation using graphics processing units. Computer methods and programs in biomedicine 98, 278–284.

Moeskops, P., Viergever, M.A., Mendrik, A.M., De Vries, L.S., Benders, M.J., Išgum, I., 2016. Automatic segmentation of mr brain images with a convolutional neural network. IEEE transactions on medical imaging 35, 1252–1261.

Navarro, F., Shit, S., Ezhov, I., Paetzold, J., Gafita, A., Peeken, J.C., Combs, S.E., Menze, B.H., 2019. Shape-aware complementary-task learning for multi-organ segmentation, in: International Workshop on Machine Learning in Medical Imaging, Springer. pp. 620–627.

Nie, D., Wang, L., Gao, Y., Shen, D., 2016. Fully convolutional networks for multi-modality isointense infant brain image segmentation, in: 2016 IEEE 13Th international symposium on biomedical

imaging (ISBI), IEEE. pp. 1342–1345.

Ourselin, S., Roche, A., Subsol, G., Pennec, X., Ayache, N., 2001. Reconstructing a 3d structure from serial histological sections. Image and vision computing 19, 25–31.

Paus, T., Collins, D., Evans, A., Leonard, G., Pike, B., Zijdenbos, A., 2001. Maturation of white matter in the human brain: a review of magnetic resonance studies. Brain research bulletin 54, 255–266.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Sanroma, G., Benkarim, O.M., Piella, G., Ballester, M.Á.G., 2016. Building an ensemble of complementary segmentation methods by exploiting probabilistic estimates, in: International Workshop on Machine Learning in Medical Imaging, Springer. pp. 27–35.

Shi, F., Fan, Y., Tang, S., Gilmore, J.H., Lin, W., Shen, D., 2010a. Neonatal brain image segmentation in longitudinal mri studies. Neuroimage 49, 391–400.

Shi, F., Yap, P.T., Fan, Y., Gilmore, J.H., Lin, W., Shen, D., 2010b. Construction of multi-region-multi-reference atlases for neonatal brain mri segmentation. Neuroimage 51, 684–693.

Sun, Y., Gao, K., Wu, Z., Li, G., Zong, X., Lei, Z., Wei, Y., Ma, J., Yang, X., Feng, X., et al., 2021. Multi-site infant brain segmentation algorithms: The iseg-2019 challenge. IEEE Transactions on Medical Imaging 40, 1363–1376.

van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., the scikit-image contributors, 2014. scikit-image: image processing in Python. PeerJ 2, e453. URL: https://doi.org/10.7717/peerj.453, doi:10.7717/peerj.453.

Wang, L., Gao, Y., Shi, F., Li, G., Gilmore, J.H., Lin, W., Shen, D., 2015. Links: Learning-based multi-source integration framework for segmentation of infant brain images. NeuroImage 108, 160–172.

Wang, L., Li, G., Adeli, E., Liu, M., Wu, Z., Meng, Y., Lin, W., Shen, D., 2018a. Anatomy-guided joint tissue segmentation and topological correction for 6-month infant brain mri with risk of autism. Human brain mapping 39, 2609–2623.

Wang, L., Li, G., Shi, F., Cao, X., Lian, C., Nie, D., Liu, M., Zhang, H., Li, G., Wu, Z., et al., 2018b. Volume-based analysis of 6-month-old infant brain mri for autism biomarker identification and early diagnosis, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 411–419.

Wang, L., Nie, D., Li, G., Puybareau, É., Dolz, J., Zhang, Q., Wang, F., Xia, J., Wu, Z., Chen, J.W., et al., 2019. Benchmark on automatic six-month-old infant brain segmentation algorithms: the iseg-2017 challenge. IEEE transactions on medical imaging 38, 2219–2230.

Wang, L., Shi, F., Yap, P.T., Lin, W., Gilmore, J.H., Shen, D., 2013. Longitudinally guided level sets for consistent tissue segmentation of neonates. Human brain mapping 34, 956–972.

Zhang, W., Li, R., Deng, H., Wang, L., Lin, W., Ji, S., Shen, D., 2015. Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. NeuroImage 108, 214–224.

Zhang, Y., Shi, F., Wu, G., Wang, L., Yap, P.T., Shen, D., 2016. Consistent spatial-temporal longitudinal atlas construction for developing infant brains. IEEE transactions on medical imaging 35, 2568–2577.

Zimmerman, J.J., Fuhrman, B.P., 2011. Pediatric Critical Care E-Book. Elsevier Health Sciences.

Appendix A. Additional Results

Experiment #	Prior	Type of Registration	Prior Pre-processing	Patch Size	Patch Stride	Data Augmentation	DSC		
							CSG	GM	WM
			Baseline - Affine label propagation						
1	/	A	/	/	/	/	0.53	0.66	0.62
			U - Net with Affine Registered Prior						
2	No	/	/	16x16x16	16x16x16	/	0.886	0.833	0.787
3	Yes	A	GB	16x16x16	16x16x16	/	0.886	0.839	0.801
4	Yes	A	GB	16x16x16	16x16x16	Yes	**0.89**	**0.84**	**0.808**
			Patch Modification						
5	Yes	A	GB	16x16x16	16x16x16	Yes	0.89	0.84	0.808
6	Yes	A	GB	16x16x16	8x8x8	Yes	**0.919**	**0.879**	**0.854**
7	Yes	A	GB	32x32x32	32x32x32	Yes	0.88	0.828	0.791
8	Yes	A	GB	32x32x32	16x16x16	Yes	0.918	0.874	0.844
			Improved Prior - Combined affine and non-rigid label propagation						
9	/	C	/	/	/	/	0.679	0.717	0.713
			U - Net with Combined Registration Prior						
10	Yes	C	GB	16x16x16	8x8x8	Yes	0.919	0.879	0.854
11	Yes	C	E + GB	16x16x16	8x8x8	Yes	**0.92**	**0.88**	**0.855**
			U - Net with No Prior						
12	/	/	/	16x16x16	8x8x8	Yes	0.918	0.879	0.853
			U - Net with Perfect Prior						
13	Yes	C	E + GB	16x16x16	8x8x8	Yes	0.929	0.925	0.927
			Multi branch U-Net						
14	Yes	/	/	16x16x16	8x8x8	Yes	0.915	0.874	0.850

Table A.13: Summary of all the experiments performed. *[A - affine; C - combined affine and non-rigid registration; GB - Gaussian blurring; E - erosion]*

Experiment #	Prior	Type of Registration	Prior Pre-processing	Patch Size	Patch Stride	Data Augmentation	ml		
							diff	diff GM	diff WM
			Baseline - Affine label propagation						
1	/	A	/	/	/	/	10.6384	29.8107	29.7237
			U - Net with Affine Registered Prior						
2	No	/	/	16x16x16	16x16x16	/	1.5834	34.2455	37.8772
3	Yes	A	GB	16x16x16	16x16x16	/	2.2129	27.3538	29.0518
4	Yes	A	GB	16x16x16	16x16x16	Yes	**1.5495**	**26.7073**	**25.6036**
			Patch Modification						
5	Yes	A	GB	16x16x16	16x16x16	Yes	1.5495	26.7073	25.6036
6	Yes	A	GB	16x16x16	8x8x8	Yes	**0.7964**	**22.0451**	**21.4974**
7	Yes	A	GB	32x32x32	32x32x32	Yes	1.8951	27.2299	28.7181
8	Yes	A	GB	32x32x32	16x16x16	Yes	1.082	22.9151	25.2725
			Improved Prior - Combined affine and non-rigid label propagation						
9	/	C	/	/	/	/	17.1405	14.0303	34.0913
			U - Net with Combined Registration Prior						
10	Yes	C	GB	16x16x16	8x8x8	Yes	0.9463	22.7446	20.5423
11	Yes	C	E + GB	16x16x16	8x8x8	Yes	**1.1092**	**22.6656**	**22.318**
			U - Net with No Prior						
12	/	/	/	16x16x16	8x8x8	Yes	0.9463	22.7446	20.5423
			U - Net with Perfect Prior						
13	Yes	C	E + GB	16x16x16	8x8x8	Yes	0.6855	12.1307	9.6185
			Multi branch U-Net						
14	Yes	/	/	16x16x16	8x8x8	Yes	0.929	23.369	23.167

Table A.14: Summary of all the experiments performed with respect to absolute volumetric difference between the total intracranial volume of segmentation and GT (column diff), difference in GM volume (diff GM) and difference in WM volume (diff WM).