

Automatic Detection of Underwater Chain Links using a Forward-Looking Sonar

Natàlia Hurtós, Narcís Palomeras, Sharad Nagappa and Joaquim Salvi

University of Girona, Edifici Politecnica IV, Campus Montilivi 17071 Girona, Spain

{nhurtos, npalomer, snagappa, qsalvi}@eia.udg.edu

Abstract—Underwater chain cleaning and inspection tasks are costly and time consuming operations that must be performed periodically to guarantee the safety of the moorings. We propose a framework towards an efficient and cost-effective solution by using an autonomous underwater vehicle equipped with a forward-looking sonar. As a first step, we tackle the problem of individual chain link detection from the challenging forward-looking sonar data. To cope with occlusions and intensity variations due to viewpoint changes, the recognition problem is addressed as local pattern matching of the different link parts. We exploit the high frame-rate of the sonar to improve, by registration, the signal-to-noise ratio of the individual sonar frames and to cluster the local detections over time to increase robustness. Experiments with sonar images of a real chain are reported, showing a high percentage of correct link detections with good accuracy while potentially keeping real-time capabilities.

I. INTRODUCTION

Chain moorings on floating structures such as floating production, storage and off-shore (FPSO) vessels are subject to harsh environmental and structural conditions that can cause wear, fatigue cracking, corrosion, distortion and other critical chain issues. Therefore, the monitoring of the chain integrity through regular and thorough inspections is of major concern as any failure can have catastrophic consequences. While the traditional methods to conduct inspections require the chain to be recovered on deck or moved ashore, in situ in-water inspection methods are improving thanks to the deployment of remotely-operated vehicles (ROVs) with optical calipers [1][2].

However, most existing solutions require prior removal of the marine growth so that the underlying chain can be properly inspected. Cleaning solutions range from manual brushing with divers, which is potentially hazardous and has an inherent depth limit, to high-pressure water systems deployed with ROVs [3]. The time spent to clean the marine growth strongly depends on the selected cleaning option, but in general is a tedious and slow task since the optical visibility drops drastically as the removed marine growth floats in the water. Indeed, considering the cost of ROV vessels, the cleaning of marine growth can be a significant fraction of the cost of a chain inspection program [3].

This work has been supported by the FP7-ICT-2011-7 project PANDORA-Persistent Autonomy through Learning, Adaptation, Observation and Re-planning (Ref 288273) funded by the European Commission and the Spanish Project ANDREA/RAIMON (Ref CTM2011-29691-C02-02) funded by the Ministry of Science and Innovation.

This work is within the context of developing a cost and time efficient solution for the cleaning and inspection of mooring chains using an autonomous underwater vehicle (AUV). In order to autonomously navigate along the chain and locate each of the links, we propose to use a high resolution imaging sonar [4][5], which delivers acoustic images at near video frame rate allowing the operation in high turbidity conditions. In this way the cleaning process is sped up while avoiding the presence of troublesome ROV cables and reducing the cost of the deploying vessel. Moreover, by producing an enhanced composition of the images gathered along the chain trajectory, the same methodology provides the means to perform a first visual inspection where it is possible to identify some major issues or locate problematic parts that need further inspection.

The paper focuses on the automatic chain link detection from Forward-Looking Sonar (FLS) imagery, which is the first step of the framework that we propose for autonomous chain cleaning and inspection. An overview of this general framework is introduced in next section. Section III presents the algorithm for individual link detection. Experimental tests and results are reported in Section IV, showing the performance of the link detection algorithm on real sonar data gathered along a chain.

II. OVERVIEW OF THE AUTONOMOUS CHAIN CLEANING AND INSPECTION FRAMEWORK

The proposed framework that we propose for the autonomous chain following and cleaning is composed of three main stages, as shown in Fig. 1. In order not only to follow the chain, but to be able to clean it appropriately, it is required to detect the position and orientation of each of the individual chain links. Therefore, the first and key stage consists in developing a robust link detection algorithm from sonar images, which is the focus of the present paper.

The detected link positions and orientations along the vehicle trajectory will then be successively incorporated as features in a global map. The mapping system follows the methodology presented in [6] substituting the visual features by link detections. When there are new observations of the same link, which are determined based on distances to previous detections, the location and orientation of the corresponding link is updated. This global mapping stage ensures the consistency of the chain, filtering out outlier detections that are not consistent with the detections already incorporated on the global map. Finally, the information of

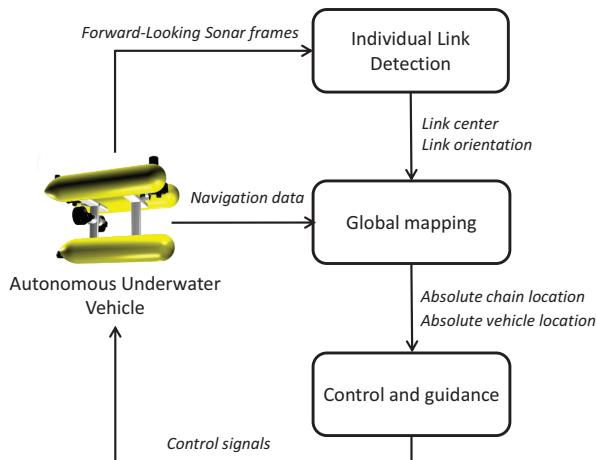


Fig. 1: Schematic of the general framework for autonomous chain following and cleaning.

the succeeding link locations on the global map is converted to control signals to guide the vehicle from link to link and perform the cleaning. The details of the cleaning procedure, by means of a water jet integrated on the AUV, fall beyond the scope of this paper.

It is worth highlighting that the presented methodology assumes the chain lying approximately on a plane, but could be extended to other configurations with the aid of a pan-and-tilt unit in order to set the sonar to the appropriate imaging configuration.

III. LINK DETECTION

While there is a large body of research on object recognition techniques on optical images, the literature is more restricted when working on sonar data, and especially under real-time constraints. Despite the recent advances on resolution of FLS, the nature of their data presents challenging conditions to the traditional detection techniques used on video images. For instance, the detection through point features, popularly used on optical images, becomes not reliable on sonar frames due to the low degree of feature repeatability [7]. A common practice in sonar object recognition is to take advantage of shadow cues [8][9]. Given that the chain links are not isolated objects, but interlaced elements, it is difficult to exploit the use of shadows as they cannot be identified distinctly. Likewise, gradient-based techniques or edge-based processing become unreliable as depending on the link's position there is a wide range of different outcomes in image intensities. There are many possible intensity transitions (from link to background, link to shadow, link to link, shadow to background...) besides possible sonar artifacts (e.g cross-talk or strong reflections) that can contribute to fragment and clutter an edge map, thus making more complex the task of identifying the link's contour.

The proposed method relies mainly on the intensities backscattered by the link itself, which are most of the times under partial occlusions due to the imaging viewpoint or

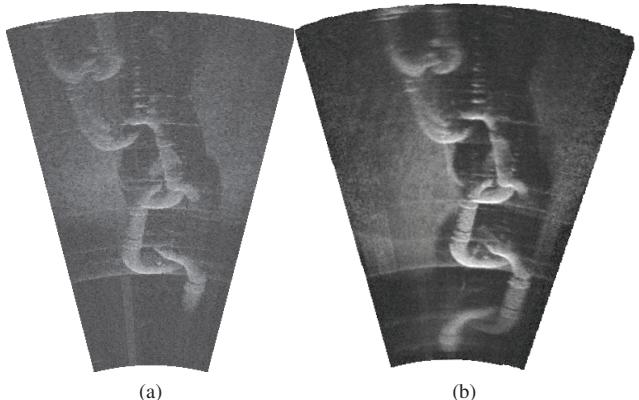


Fig. 3: (a) Example of a single sonar frame. (b) Enhanced frame computed by registering $n = 3$ consecutive frames.

actual objects occluding parts of the chain link, such as other links or marine growth. We have approached the problem as a pattern matching using normalized cross-correlation of local templates, which allows us to detect portions of the link while others are not visible. Those detections are then robustly clustered and related according to their known dimensions and spatial location to finally identify the presence of a link and provide an estimation of its center and orientation.

The general flow of the detection algorithm can be seen in Fig. 2. In the next sections each of the steps will be detailed.

A. Image enhancement

Rather than working on a frame-by-frame basis the proposed system takes advantage of the sonar high frame rate and uses registration within a small window of n consecutive frames to produce an intermediate image of increased signal to noise ratio (SNR). The registration process considers an approximated FLS projection model [10] and estimates the frame's relative displacements and rotation using Fourier-based techniques [10]. Fig. 3 shows an example of one individual sonar frame and the enhanced image computed with a window of $n = 3$ neighbours. The fused image is composed by averaging the intensities at each point thus reducing the noise and the incidence of spurious artifacts present in the individual frames. Hence an enhanced image, on which the rest of the processing pipeline will be conducted, is provided at every n sonar frames. Additionally, the relative displacement and rotation between every pair of frames is stored to be used in later steps.

B. Pattern matching

Working with the enhanced images, a template matching process can be applied with higher reliability. Given that occlusions are frequent, we treat the detection problem as a template matching in a local fashion in order to identify separately the location of different parts of the link (i.e, the four corners plus the long side segment).

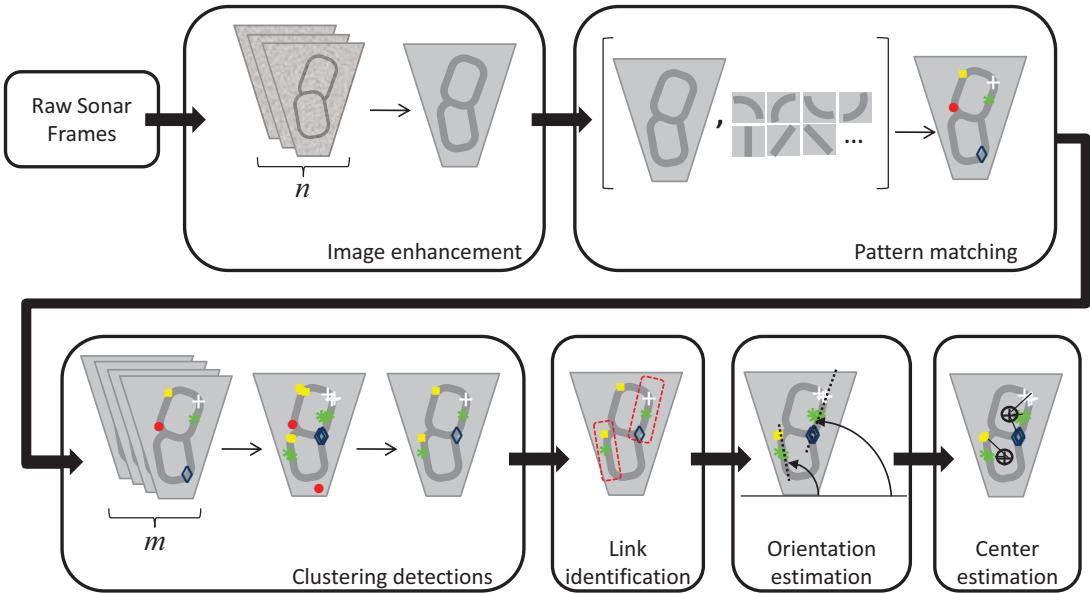


Fig. 2: Schematic of the link detection algorithm.

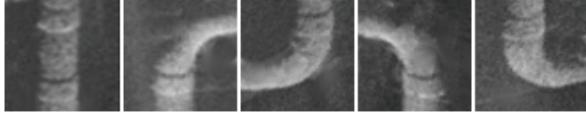


Fig. 4: Example set of basic local templates for a studless chain. For each template two more templates are generated by rotating several degrees to each side

Local templates of the chain are extracted from any available sonar image of the chain gathered in previous inspections. For a studless chain, a full set of templates consists of 5 local patches centered at the link's four corners and one centered on the longest side of the link (see Fig. 4), each of one together with two more corresponding templates rotated several degrees at each side. It is not required to use a more extended set of templates covering a finer range of orientations since the aim of this step is just to identify the position and type of different link parts. The actual estimation of the link orientation will be handled in a later stage. Thus, for example, any of the three different upper left corner templates will output a stronger correlation on the position of an upper-left corner than any of the other patterns, even if the rotation is slightly different.

The template set is strongly related to the morphology of the chain and must be tailored to each particular chain typology. For instance, if the focus of inspection is a chain of stud links, the straight segment template could be replaced for two extra templates containing the t-shaped connection in the center of the link. However, knowing the settings in which the images were gathered (initial and end range of the sonar frame) a given set of templates can be scaled accordingly to be used on other datasets gathered on the same chain but at different sonar configurations

Therefore, for a given image, we perform normalized cross-correlation for each of the local template patterns:

$$Corr(u, v) = \frac{\sum_{x,y} [I(x, y) - \bar{I}_{u,v}][T(x - u, y - v) - \bar{T}]}{\sqrt{\sum_{x,y} [I(x, y) - \bar{I}_{u,v}]^2 \sum_{x,y} [T(x - u, y - v) - \bar{T}]^2}} \quad (1)$$

where I is the image, T is the template, \bar{T} is the mean of the template intensities and \bar{I} the mean of the image intensities under the template. For each template class (i.e. upper left, lower left, upper right and lower right corners and straight segment), we keep the location of the maximum correlation $Corr(u, v)$, thus identifying in each image one and only one detection of each class. Given that many links would be usually imaged in a single frame, correlation of a template gives different areas with high values. However, in order to maximize the robustness of the detections only the strongest one is kept at each image.

Notice that the 5 template detections can be located on the different parts of the same link, distributed on different links or even in locations where there is no link at all (outlier template detections caused by miscorrelations or the lack of links in the image). The most common case is that different template detections are spread in several links, as the right and the left sides of a particular link are not usually observed at the same time due to the inclination caused by the link interlacing (see Fig. 5).

C. Clustering of detections

In order to increase the robustness of the detection mechanism and discard outliers, template detections are accumulated over time. A buffer of m consecutive enhanced images (I_1, I_2, \dots, I_m) is established and their individual template detections are accumulated by referencing all of them with respect the sonar origin of I_1 . Transformations needed for

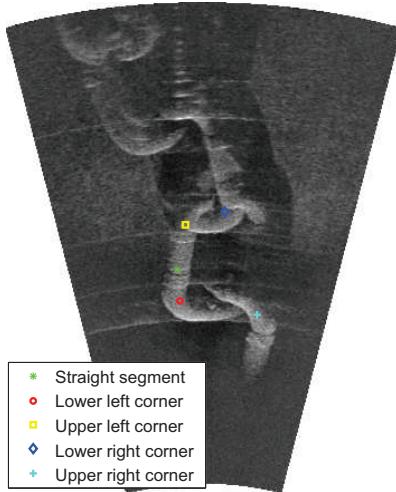


Fig. 5: Example of the detections obtained by cross-correlation of the 5 different template classes. Only the maximum of each class is retained. In this case detections are spread over 3 different links.

bringing each detection from I_i to I_1 , are obtained from the registrations computed in the image enhancement stage. m corresponds to the number of enhanced images generated over a displacement equivalent to the length of a link. Equation (2) shows an approximated rule to determine m according to the sonar acquisition frame rate (s_{fps}), vehicle velocity (v), link length (h) and number of registered raw frames for each enhanced image (n):

$$m = \frac{s_{fps} \cdot h}{v \cdot n} \quad (2)$$

Therefore, every m images the algorithm will accumulate 5-by- m detections. As pointed out earlier, those detections might be spread in different locations, but even the ones that have targeted the same corner or same link segment might have some position variability. This is due to slightly different positions of the correlation peak or due to the accumulated error in the transformation that brings the detection to the first image of the buffer. It is then required to cluster those detections by classes in a nearest-neighbour fashion. For a given class, detections will be considered to be part of the same group if they are within less than half link distance ($h/2$) of other detections of the same class.

Groups with less than $m/3$ detections are discarded. In this way, we force the detections to have enough support over time (i.e. should be detected at least over a third of the different images) to be considered valid detections. Note that, the number of links that can be detected in a single image is limited by this threshold and thus can happen that the algorithm only keeps valid detections on one of the links, despite more links might be present in the image. This favors the robust detection of the link or links that are better observed along the m images, rather than the identification of all of them. As the vehicle is moving along the chain, these

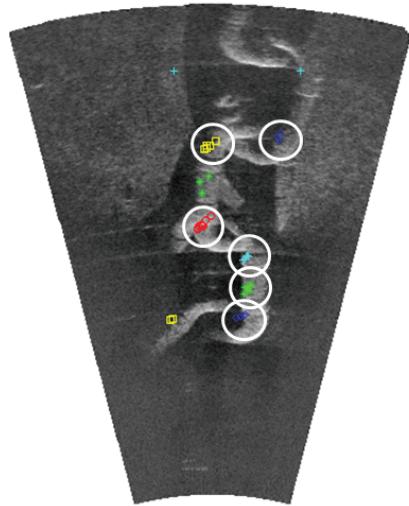


Fig. 6: Example of established clusters (circled in white) after the accumulation of detections over a buffer of $m = 9$ images. Notice that several detections, even correct ones, are discarded as they are not part of a group with enough support (in this case, less than 3 detections).

links will start to drop out of the sonar's field of view and links that were imaged at further ranges (and therefore lower resolutions) will be imaged later at closer ranges. Hence if a link has not had any detection in a given image due to a poor insonification viewpoint it has chances to be detected more robustly in later images.

Fig. 6 shows an example of detections accumulated along $m = 9$ enhanced images and the clusters that have been identified.

D. Link identification

Clustered detections that were tackled as valid must be associated into groups belonging to the same link. To identify which detections are part of the same link we make use of an heuristic that explores a series of possibilities according to the cluster's spatial location and known link dimensions.

The applied heuristic assumes that given the main direction in which the chain is deployed, the links can not have sudden orientation change of more than 90 degrees from one link to the next one. This is a reasonable assumption given that mooring chains are typically subject to tension and even they can have curved trajectories, those vary smoothly with most of the links following an orientation around the main chain deployment direction.

Due to link interlacing, alternate links have different inclinations across the main chain axis, making elements of the same side easier to be seen at the same image. Therefore the heuristics sweeps the image in the chain direction (here considered vertically) for the combination of elements (either two or three) of the same side, rather than looking for side-way combinations such as [lower-left corner/right-left corner]. Thus, the heuristic mechanism looks for combinations or sub-combinations of clusters corresponding to the follow-

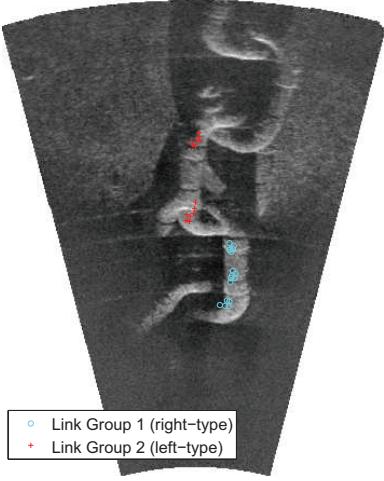


Fig. 7: Link identification performed on the clusters of Fig. 6. Two different groups have been identified.

ing sequences: [lower-left corner/straight segment/upper-left corner] or [lower-right corner/straight segment/upper-right corner].

Two elements are considered to be part of the same group if they are within half-link distance or within link distance, depending on the element class. Two corners of the same side are a valid combination if they are within link distance, while a corner and a straight segment of the same side must be closer than half-link distance to be considered part of the same link. In order to compute the distance, the centroid location of the clusters is used.

Fig. 7 shows an example of the link identification performed on the clusters of Fig. 6 where two different groups have been identified. The first one is a two-element combination of left-side groups ([lower-left corner/upper-left corner]) and the second is a full combination of right-side groups ([lower-right corner/straight segment/upper-right corner]). Notice that one of the lower-right corner clusters identified in Fig. 6 has not been grouped as no possible combination has been found around thus yielding weak evidence of the presence of a link in that location.

E. Orientation estimation

Orthogonal regression is used to fit a line through the points composing the different groups of a given link. The orientation of this line with respect to the image x axis gives an estimate of the link's orientation. Fig. 8 shows examples of orientation estimation for links identified out of 2 and 3 class groups.

F. Center estimation

Given an estimation of a link's orientation α , the theoretical center position $[c_x, c_y]$ can be projected from each of the detected points p as:

$$\begin{bmatrix} c_x \\ c_y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & p_x \\ \sin(\alpha) & \cos(\alpha) & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ 1 \end{bmatrix} \quad (3)$$

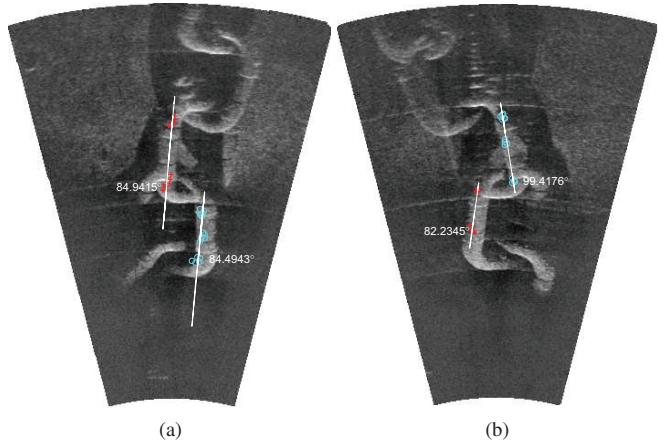


Fig. 8: Example of different orientation estimations.

where $[t_x, t_y]$ are the displacements of the link center with respect the theoretical location of point's p class. Fig. 9a shows the projection of the center locations from each of the detections identified in Fig. 7.

This two-dimensional center projection is just an approximation as due to the chain interlacing the links are not lying horizontally on the plane. Moreover, the correlation maximum detected in the pattern matching step (i.e points p) may not correspond to the exact location of the link part. To handle this uncertainty, each projected center is converted to a gaussian probability distribution function (PDF) of $\sigma = ts/2$, where ts is the template size. Then, the final center position is estimated as the maximum of the probability map generated from the addition of all gaussian distributions of the link's group (see Fig. 9b).

Apart from a more robust estimation of the center location, this provides us a simple way to extract an estimation of the confidence of the center position. The maximum of the combined PDF is cut at its half height and the variance in every axes direction provides an estimate of the center location uncertainty. This measure can then be converted to meters and used within the global mapping stage of the framework.

IV. EXPERIMENTS AND RESULTS

In order to test the link detection algorithm, three datasets have been acquired by using Girona500 AUV [11] equipped with an ARIS FLS from Sound Metrics [4]. The vehicle has been teleoperated inside a water tank containing a studless chain composed of 13 links 0.7 m long each (see Fig. 10). FLS imagery has been collected while navigating all over the chain through different viewpoints. In datasets 1 and 2 the vehicle starts around the first link and follows along the chain with slightly different trajectories. In dataset 3 the vehicle begins at the other end of the chain, going from the last link to the first.

The ARIS sonar was configured to image a window from 1 to 4 meters ahead of the vehicle, thus being able to image 3 or 4 links in every frame. The frame rate of acquisition

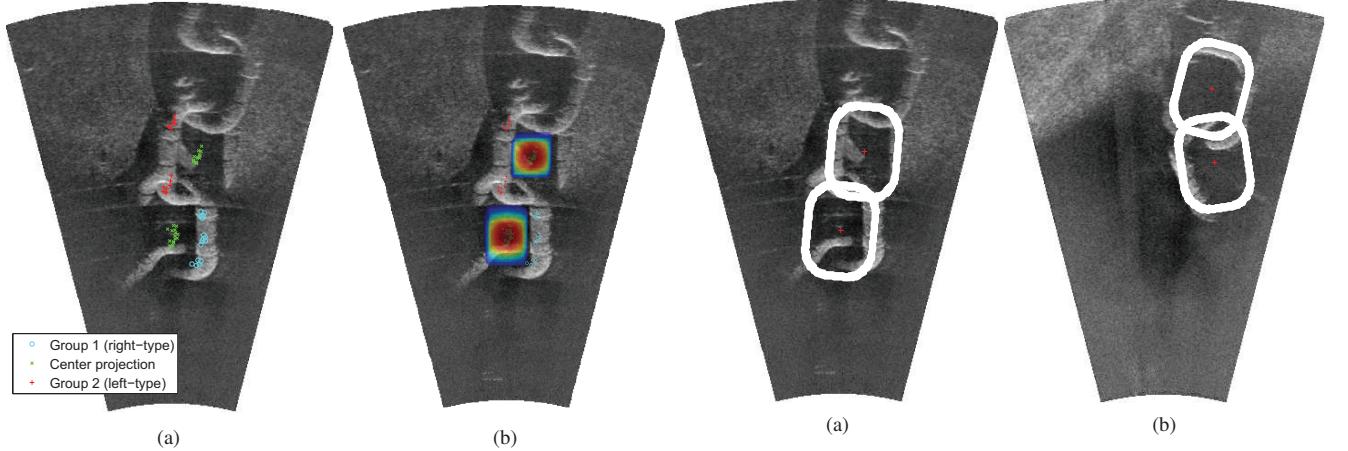


Fig. 9: (a) Projections of the theoretical center locations from each of the groups detections. (b) Overlay of the probability map generated as the sum of individual gaussian distributions for each projected center. The link center is estimated as the maximum in each combined pdf.



Fig. 10: Chain deployed in the water tank for the experiments.

was set at 9 frames per second and the neighbourhood for the image enhancement stage to $n = 3$. According to Equation (2), as the vehicle velocity was around 0.25 m/s, m has been established to 9 images. Hence, inside our general framework, the algorithm would output estimations of link positions approximately every meter (3 seconds). The employed set of templates extracted from several images can be seen in Fig. 4.

Fig. 11 shows examples of final detections along the chain, overlying a link pattern centered at the estimated center location and rotated according to the estimated link orientation angle.

Tables I to III summarize the obtained results for each of the datasets. We report the percentage of detected links with respect the total number of chain links (13 in this case), the number of outlier detections and the mean orientation and center error of the detections. In order to have a ground truth to compare the accuracy of the estimations we have manually identified the center and orientation of all the links detected by the algorithm.

The first thing to be highlighted is that despite the imposed restrictions that in most of the cases limit the number of the

Fig. 11: Examples of final link detections, with a link overlay centered at the estimated center location and rotated according to the estimated link orientation angle. The lower link on (b) shows an example of poor accuracy on the rotation estimation.

TABLE I: Results summary dataset 1

| | |
|-----------------------------------|------------------------|
| Percentage of detected links | 92% |
| Number of outlier link detections | 0 |
| Mean orientation error | $6.1 \pm 4.1^\circ$ |
| Mean center location error | $3.6 \pm 2.8\text{cm}$ |

detections to just one or two links on the individual images, very few links remain undetected all along the dataset. The worse case is found in dataset 2, where 2 links remain undetected giving a percentage of detected links of 84%. It must be pointed out that almost all cases of undetected links are located at the beginning of the trajectory in close sonar ranges. That implies that when the vehicle advances those links quickly drop out of the sonar field of view thus appearing in less images and having less chances of being detected. Moreover, provided that non-detections are isolated cases, the global mapping stage that follows the link detection can identify the lack of a link in the global chain trajectory and interpolate its position.

As a result of using enhanced images and requiring a

TABLE II: Results summary dataset 2

| | |
|-----------------------------------|-------------------------|
| Percentage of detected links | 84% |
| Number of outlier link detections | 2 |
| Mean orientation error | $6.12 \pm 5.95^\circ$ |
| Mean center location error | $5.14 \pm 5.3\text{cm}$ |

TABLE III: Results summary dataset 3

| | |
|-----------------------------------|--------------------------|
| Percentage of detected links | 92% |
| Number of outlier link detections | 3 |
| Mean orientation error | $7.03 \pm 5.21^\circ$ |
| Mean center location error | $5.88 \pm 4.90\text{cm}$ |

TABLE IV: Results summary dataset 1 (Over individual frames)

| | |
|-----------------------------------|--------------------------|
| Percentage of detected links | 69% |
| Number of outlier link detections | 6 |
| Mean orientation error | $14.34 \pm 8.23^\circ$ |
| Mean center location error | $9.18 \pm 7.66\text{cm}$ |

strong detection support over time, the number of outlier detections over all the individual images is not significant. Outlier cases generally arise in those locations where a link shadow creates a curved area similar to a corner template and hence if the viewpoint does not change sufficiently they are consistently detected along several frames. However, that happens occasionally and it can be also handled by the subsequent global mapping stage. Known link dimensions can be used to enforce consistency between the link detections on the map and the upcoming ones thus excluding those outliers that are far off the chain trajectory.

The accuracy of the obtained detections, specially the link center location, is reasonably good given the sonar resolution and the two-dimensional projection assumption. Besides, the final planning of the link cleaning trajectories can take into account the uncertainty of the detections and adjust the trajectories to ensure that a wider area is covered by the water jet.

It is worth to emphasize the benefit of working with the enhanced images rather than the raw sonar frames. Table IV summarizes a test performed on the raw sonar frames of dataset 1 where it can be clearly seen that the number of outliers rises considerably and the accuracy drops significantly.

Regarding computational time, the algorithm has been implemented in MATLAB and tested on a Intel Core 2 Duo at 3.0Ghz. For the selected sonar configuration and parameters n and m , new link detections are computed approximately every 8 seconds while the real-time throughput should be around 3 seconds. The most time-consuming step is the computation of the frame registrations that are required to compute the enhanced images and also to accurately accumulate the different detections over the image buffer. For the test datasets, composed of images around 350x450 pixels, a pairwise registration takes 0.2 seconds. Therefore the generation of each enhanced image takes around 0.6 seconds, which leads to more than 5 seconds to generate the m enhanced images that are accumulated. However, this time is a reasonable amount considering a MATLAB implementation. Hence, with a more efficient implementation the algorithm has high potential to run in real-time.

V. CONCLUSIONS

We have developed a robust algorithm to automatically detect chain links from sonar imagery in the context of an autonomous chain cleaning and inspection scenario. The algorithm takes advantage of the high frame rate of new 2D-FLS to enforce robustness over time. It includes an image enhancement procedure by registration of consecutive

frames, a correlation of local templates to improve handling of occlusions, and a simple heuristics mechanism on a buffer of clustered detections in order to determine link position and orientation. The performance of the link detection has been tested on real sonar data, proving to be robust enough to handle the challenging characteristics of FLS images, giving a low rate of outliers and reasonable accuracy while maintaining the capability to work in real time.

Further work will concentrate on integrating the algorithm inside the general framework presented on Fig. 1 and developing the rest of the stages towards an autonomous chain cleaning and inspection solution.

REFERENCES

- [1] A. Hall, "Cost effective mooring integrity management," in *Offshore Technology Conference*, 2005.
- [2] C. Morandini and F. Legerstee, "Consistent integrity of mooring systems," *ISOPE, Osaka, Japan*, 2009.
- [3] Noble Denton Europe Limited, "Research report 444 floating production system," Health and Safety Executive (HSE), Tech. Rep., 2006.
- [4] "Sound Metrics Corp." [Online]. Available: www.soundmetrics.com
- [5] "Blueview Technologies Inc." [Online]. Available: www.blueview.com
- [6] N. Palomeras, S. Nagappa, D. Ribas, N. Gracias, and M. Carreras, "Vision-based localization and mapping system for auv intervention," in *OCEANS'13 MTS/IEEE*, 2013.
- [7] N. Hurtos, S. Nagappa, X. Cufí, Y. Petillot, and J. Salvi, "Evaluation of registration methods on two-dimensional forward-looking sonar imagery," in *OCEANS'13 MTS/IEEE*, 2013.
- [8] Y. Petillot, Y. Pailhas, J. Sawas, N. Valeyrue, and J. Bell, "Target recognition in synthetic aperture and high resolution side-scan sonar," in *European Conference on Underwater Acoustics, ECUA*, vol. 10, 2010.
- [9] E. Hasanbelliu, J. Principe, and C. Slatton, "Correntropy based matched filtering for classification in sidescan sonar imagery," in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*. IEEE, 2009, pp. 2757–2762.
- [10] N. Hurtos, X. Cufí, Y. Petillot, and J. Salvi, "Fourier-based registrations for two-dimensional forward-looking sonar image mosaicing," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, oct. 2012, pp. 5298 –5305.
- [11] D. Ribas Romagós, N. Palomeras Rovira, P. Ridao Rodríguez, M. Carreras Pérez, and A. Mallios, "Girona 500 auv: From survey to intervention," *IEEE/ASME Transactions on Mechatronics*, 2012, vol. 17, núm. 1, p. 46-53, 2012.