IMAGE MOSAICKING FOR ESTIMATING THE MOTION OF AN UNDERWATER VEHICLE

Rafael García, Xevi Cufí and Lluís Pacheco

Computer Vision and Robotics Group
Institute of Informatics and Applications, University of Girona
Edifici Politècnica II, Escola Politecnica Superior
17071 Girona, Spain
e-mail: {rafa,xcuf}@eia.udg.es

Abstract: A composite image constructed by combining a set of smaller images is known as mosaic. Mosaics of the ocean floor are very useful in undersea exploration, creation of visual maps, navigation, etc. A feature-based mosaicking method is proposed, based on textural parameters of certain parts of the image. Textures significantly help in the identification of given features of the image. The tracking of these features over time solves the matching problem in consecutive frames. The correspondence is established with the aid of a cross-correlation algorithm, applied to the colour components of the image in the HSI space. Once the correspondence has been found a displacement vector is obtained relating the features of two images of the sequence. The motion parameters between consecutive frames are estimated through an error minimisation technique. Once the best transformation between two frames has been found, images are warped together composing the mosaic, and information about the vehicle motion is recovered. The experimental results on real images show the effectiveness of the proposed method.

Keywords: Computer Vision, Motion estimation, Autonomous vehicles, Image registration, Robot navigation, Marine Systems.

1. INTRODUCTION

A *mosaic* is a set of images which have been combined into a single and larger composite image (Marks, *et al.*, 1995). Video mosaics of the ocean floor are very useful for many applications, including undersea exploration, creation of visual maps, navigation, pipe inspection, etc. However, the application of standard computer vision techniques for underwater imaging involves dealing with different problems. Underwater images often lack distinct features (e.g., points, lines or contours) that are commonly exploited in terrestrial vision systems for tracking, positioning, navigation, etc. Moreover, the range is limited and the need for artificial light introduces many new properties to the image, such as low contrast and non-uniform illumination. Quite often, small observable particles suspended in the water show up as marine snow making feature extraction difficult. For all these reasons, the processing of underwater images appears as a challenging area of research.

For ocean floor mosaics, the individual images forming the mosaic are usually obtained by setting a camera on a ROV (Remotely Operated Vehicle) or AUV (Autonomous Underwater Vehicle). The camera is typically looking down to the bed of the sea, and the acquired images cover a small area of the ocean floor. By using this technique, the position and orientation of the underwater vehicle can be calculated by integrating the motions from one image to the next one (Xu and Negahdaripour, 1997; Negahdaripour, *et al.*, 1998a).

If our aim is simply to construct a map of the bed of the sea, other methods are found in the literature. Side scan sonar techniques have been applied to construct maps of the sea floor, see (Langer and Hebert, 1991). However, the reconstruction accuracy is poorer than that provided by optical devices. Interesting results in column-based mosaicking can be found in the literature (Marks, *et al.*, 1995) by using a constrained four-parameter semi-rigid motion model. Moreover, unconstrained image mosaicking can be obtained applying smoother-follower techniques to reduce image alignment errors within the mosaic (Fleischer, *et al.*, 1996). Some of the projects involving experimental research in underwater imagery perform mosaicking in real time (Negahdaripour, *et al.*, 1998b), without the need of any additional inertial systems, since updating the position of the vehicle can be achieved solely from visual information. Figure 1 shows how the motion of an underwater vehicle can be estimated from a sequence of images.

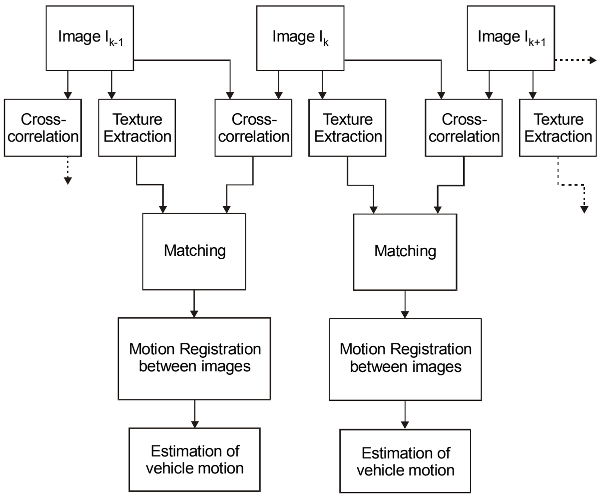

Fig. 1: Scheme of the algorithm for locating correspondence points between pairs of consecutive images

2. BACKGROUND

Mosaicking techniques can be divided into two categories: feature-based methods, and featureless methods. The first assume that feature correspondences between image pairs can be established, and utilise these correspondences to find transforms which estimate the motion parameters between image pairs. This computation of the motion parameters is known as *registration* in the literature (Szeliski and Kang, 1995). The second method finds the transforms for image registration by minimising a sum of squared difference (SSD) function that involves some parameters. When applying these methods to underwater images, the extraction of features appears as a difficult task. Methods in this category typically assume that the change from one image to another be small. In order to find the apparent motion from one image to another, good initial guesses for the parameters of the transform are required as initial values to find the optimal solution. Failing that, the parameter estimate process could fall into local minima (Szeliski, 1994).

In this paper, a feature-based technique is proposed. The algorithm is applied to a sequence of images at a frame rate of 25 images/second. In this way, the smoothness requirement that pushes through the matching procedure is accomplished. Our aim is to use a projective transform-based method for creating underwater mosaics since no assumptions have to be made in the camera motion.

3. THE ALGORITHM: MOSAIC CREATION

In this section the proposed algorithm for creating mosaics from a sequence of images is exposed. The creation of the mosaic is accomplished in three stages: feature selection and matching within images, registration and mosaic blending.

3.1 Feature matching

The first step towards the estimation of the image registration parameters consists in finding feature correspondences. This is referred to as the matching problem, which is normally solved in aerial images through the extraction of the high frequencies of the image, discovering the outline of the objects present in the scene. However, in underwater applications, the nature of acquired images presents serious difficulties when trying to match image corners or contours. Hence, a feasible alternative is to match highly textured patches corresponding to given regions of the ocean floor.

Our correspondence technique is a mixture of texture-based operators and correlation-based matching procedure. Large regions of the images are used to find the matches. Thus, a significant percentage of the image must change for correspondence to fail.

A colour camera and a standard frame-grabber compose the imaging system of our underwater robot (Amat, *et al.*, 1996). For every new image in the sequence (I) a colour conversion is performed, extracting the hue, saturation and intensity components of the image. A set of texture parameters is applied independently to every component, searching for distinguishable features in the image. Once the set of candidate patches are defined, a small window W_i delimits the neighbourhood of every patch, and the grey-level of this window for the components hue, intensity and saturation are stored. When the next image is acquired (I'), a new measure of textures is performed in the vicinity of the initially defined windows for every component. In order to improve the matching, a cross-correlation is carried out by taking the grey-level sub-image of each and every patch and comparing it with the new image. An

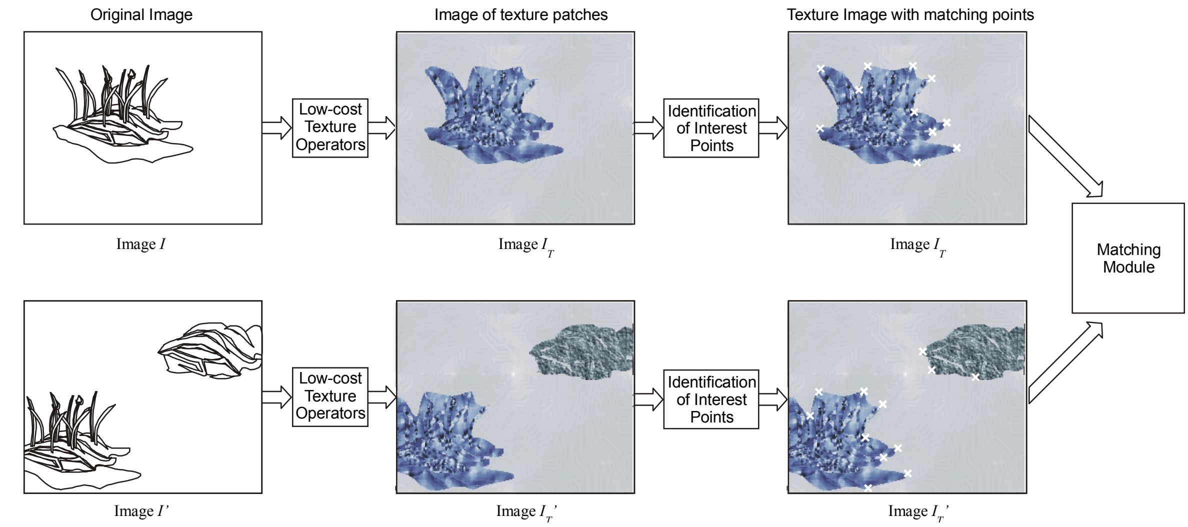

Fig. 2. Scheme of the algorithm for locating points of images I_k and I_{k+1} in the coordinates system of image I_{k-1}.

attempt is done in image *I'* for matching the previously selected features, by using both the texture-parameters and cross-correlation criteria. Figure 2 shows the scheme of the feature extraction procedure.

The textural features used in this implementation have been chosen for their suitability for underwater imaging and their low cost in terms of computing complexity. A brief explanation of their main characteristics is given below. For a more detailed description see (Casals, *et al.*, 1992).

- *Straightness*: it defines the density of straight lines in a region.
- *Granularity*: this parameter quantifies the presence of a high density of non-concatenated gradients in a given patch.
- *Blurriness*: it measures intensity changes in any direction within a region. A high value of this parameter implies that smooth changes are present.
- *Discontinuity*: this parameter measures the number of line discontinuities.
- *Abruptness*: it indicates sudden changes of the direction of the lines within a region.

Once the matching has been established a displacement vector is obtained for every patch in the first image. This vector relates the coordinates of the same texture-feature in both images. The assumption of large overlap of image contents between the two frames can be used to significantly reduce the computational burden of the matching. This is achieved by limiting the search area in the second image (*I'*). It is worth noting that image motion that cannot be described by simple translation causes the displacement vectors to be warped. Finally, the matching module provides a list of coordinates of corresponding patches. Due to the error prone nature of the matching process, it is likely that a number of point correspondences will not relate to the same 3D feature. For this reason, the next subsection is dedicated to the robust estimation of the motion parameters taking into account the existence of mismatches.

3.2 Global registration and mosaicking

This module has the goal of estimating the motion parameters between consecutive frames. Starting from the list of correspondences, a transformation matrix relating images *I* and *I'* has to be found. Projective transforms are commonly used since they permit arbitrary camera motions: translations, zooming, rotation around the optical centre, panning and tilting. The only constraint in mosaicking applications is that the scene being recorded must be planar. A projective transform depicting the inter-frame motion can be represented by a *homography*, a linear function of projective image coordinates (Szeliski, 1994):

$$\mathbf{u'} = \mathbf{M}\mathbf{u} \qquad (1)$$

where **u'** represents the image coordinates of the projection of a 3D point P in the image plane at time instant *k*, and **u** represents the coordinates of the projection of the same 3D point as projected the previous time instant *k-1*.

Expanding eq. (1) the following equation is obtained:

$$\begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} \cong \begin{bmatrix} m_{11} & m_{21} & m_{31} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \qquad (2)$$

where (x_i', y_i') and (x_i, y_i) denote a correspondence point in the present (*I'*) and previous image (*I*), respectively; $m_{11}, m_{12}, \ldots, m_{32}$ are 8 parameters that determine the projective transform; and $\cong$ indicates equality up to scale. As we said before, the image registration process consists in finding this 8-parameter set, which is achieved through a least squares iterative process. Once the best transformation **M** has been found, images I and I' can be *warped* together.

In order to construct the mosaic, the first image in the sequence is selected as a base frame. The mosaic coordinate system is placed at the origin of this reference frame. When a new image has to be added to the mosaic, matrix **M** provides its best fitting with respect to the previous image (that at the beginning of the process will be the reference one). For every image, the mosaic is only updated within the regions where no information existed before. On the other hand, all the images in the sequence are taken into account in order to keep a smooth track of the matching points. However, the mosaic is updated with a new image only every 10 frames. In this manner, only the first measured value of a pixel updates the mosaic. This is known as *use-first* method (Gracias and Santos-Victor, 1998). It has the drawback of the impossibility of removing transient data, such as a moving fish, but simplifies to a large extent the mosaic construction. We plan to implement other techniques like a median-based method, which effectively will remove moving objects whose intensity patterns are quiescent for less than half of the frames.

4. RESULTS

An experiment on real data has been performed, in order to check the suitability of the described texture parameters, as well as cross-correlation estimation for the processing of underwater images. The test sequence consists of a set of 180 underwater images.

Figure 3 shows a colour image of the sequence and the result of performing a contour detection on the luminance component. These images have been taken from our underwater robot, and belong to the image sequence where the algorithm is to be tested.

If we were to analyse directly the image of figure 3(b), it would be a difficult task to find a contour separating the main regions of the image. By applying to this image a set of texture parameters, images like the one shown on figure 4(a) can be obtained. In this image the texture components (*texels*) of one of the images is shown. The result of finding the contours from image 4(a) is illustrated in figure 4(b). It can be observed how there is a contour separating the two main regions of the image. The points in the border of both regions provide considerable information about motion. For this reason cross-correlation is performed for selected points over this edge line. Figures 4(c) and (d) illustrate the result of performing the same operation on some other image of the sequence. This time the figure shows the result of applying the texture operator over the hue component of the image.

(a)

(b)

Fig. 3: (a) Original colour image, and (b) extracted contours directly obtained over the luminance component.

It should be noted that all the texture parameters described in the previous section are applied to the three image components hue, saturation and intensity, and then, the main contours of the image are obtained. In this way, the candidate points for computing the cross-correlation are selected only from the regions providing more reliable contrast on the image. Figure 5 illustrates the result of the cross-correlation-based matching over two non-consecutive images of the sequence.

The matching is performed for every image in the sequence, applying a smoothness constraint which facilitates the correspondence. In this way, the image features can be tracked over the sequence. However, the mosaic is not updated with every new image of the sequence, but only when significant information can be added to the composite mosaic, speeding up the processing time. Figure 6 shows the result of the resulting mosaic with updates every 10 images. The results here reported concern some preliminary tests to check the ability of the algorithm to recover information about the motion of the vehicle in different underwater situations. The extension of this method to more complex contexts, such as more uniform images including algae, sand, etc., will show the accuracy of this approach to be used as a position sensor in lieu of inertial-based positional systems.

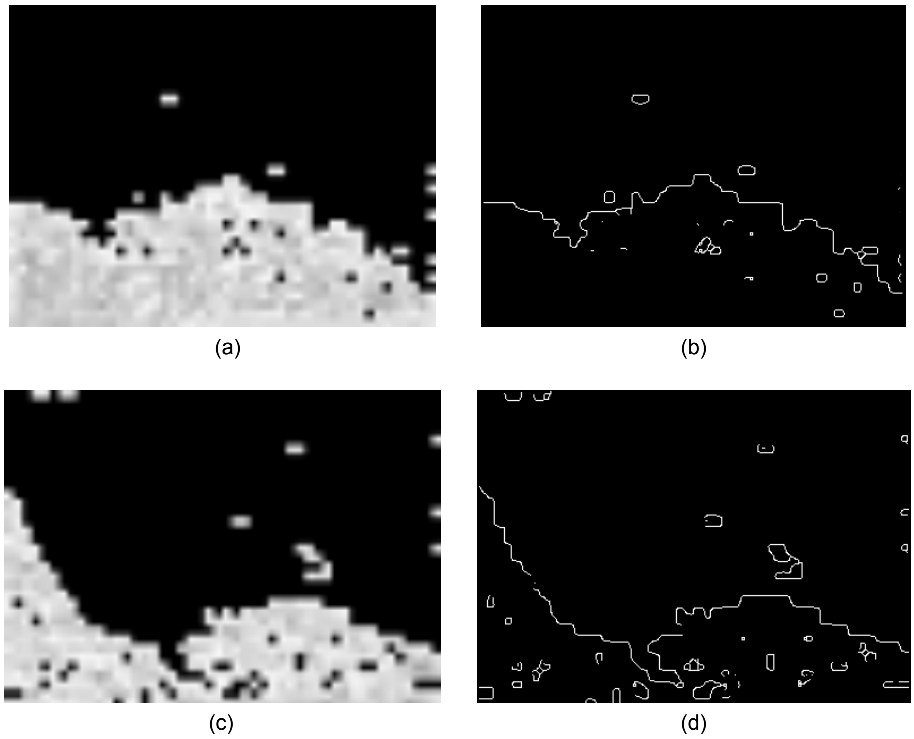

Fig. 4: (a) Example of lineality textural feature obtained over the luminance component; (b) contours obtained from (a). Cross-correlation is performed for selected points over the resulting contours; (c) Example of lineality texture parameter obtained over the hue component on another image; (d) contours obtained from (c).

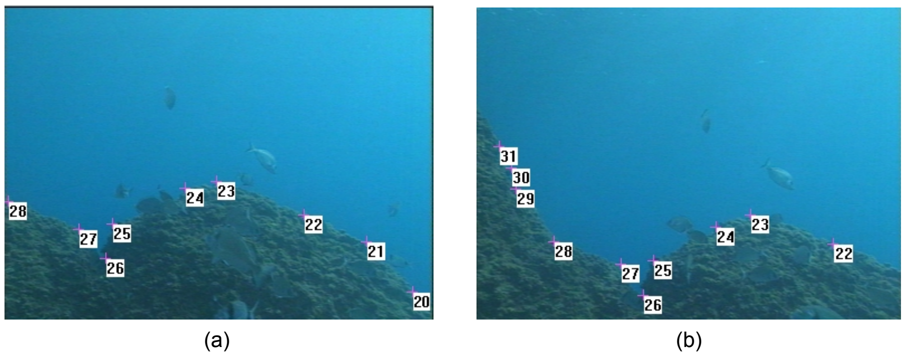

Fig. 5: Matched points between non-consecutive images of the sequence

5. CONCLUSIONS AND FURTHER WORK

A technique for creation of automatic video mosaics of the ocean floor has been presented. Our approach uses the image textures over specially selected points to establish the correspondences from one image to the next one. Registration between frames is computed for every new image of the sequence, but the mosaic is updated every 10 images, reducing the cycle time of the algorithm.

Accumulation of image alignment errors have been detected as the mosaic increases in size. In the future, optimal estimation techniques will be investigated to reduce these errors. Moreover, when moving objects are present in the scene, the registration between images can suffer slight alterations. It seems that the use of a temporal median filter on the overlapping pixels of the mosaic would improve the registration phase.

REFERENCES

Amat, J., J. Batlle, A. Casals and J. Forest (1996). GARBI: A Low Cost ROV, Constraints and Solutions. In *Proceedings of the 6th IAPR Underwater Robotics*.

Casals, A., J. Amat, and A. Grau (1992). Texture parametrizacion method for image segmentation. In *Proceedings of the Second European Conference on Computer Vision*, pp. 160–164.

Fleischer, S. D., H. H. Wang, S. M. Rock, and M. J. Lee (1996). "Video Mosaicking Along Arbitrary Vehicle Paths", in *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology*, pp. 293–299.

Gracias, N. and J. Santos-Victor (1998). Automatic mosaic creation of the ocean floor. In *Proceedings of the OCEANS Conference*, **vol. 1**, pp. 257–262.

Langer, D. and M. Hebert (1991). Building qualitative elevation maps from side scan sonar data for autonomous underwater navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

Marks, R.L., S.M. Rock, and M.J. Lee (1995). Real-time video mosaicking of the Ocean floor. *IEEE Journal of Oceanic Engineering*, **vol. 20**, no.3, pp. 229–241.

Negahdaripour, S., X. Xu, A. Khamene and Z. Awan (1998a). 3D motion and depth estimation from sea–floor images for mosaic–based station–keeping and navigation of ROVs/AUVs and high–resolution sea–floor mapping. In *Proceedings of the IEEE Workshop on Autonomous Underwater Robots*, pp. 191–200.

Negahdaripour, S., X. Xu and A. Khamene (1998b). A vision system for real–time positioning, navigation and video mosaicing of sea floor imagery in the application of ROVs/AUVs, in *Proceedings of the 4th IEEE Workshop on Applications of Computer Vision*, pp. 248–249.

Szeliski, R. and S.B. Kang (1995). Direct methods for visual scene reconstruction. In *Proceedings of the IEEE Workshop on Representation of Visual Scenes*, pp. 26–33.

Szeliski, R. (1994). Image mosaicing for tele-reality applications. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*, pp. 44–53.

Xu, X., and S. Negahdaripour (1997). Vision-based motion sensing from underwater navigation and mosaicing of ocean floor images, in Proceedings of the *MTS/IEEE OCEANS Conference*, **vol. 2**, pp. 1412–1417.

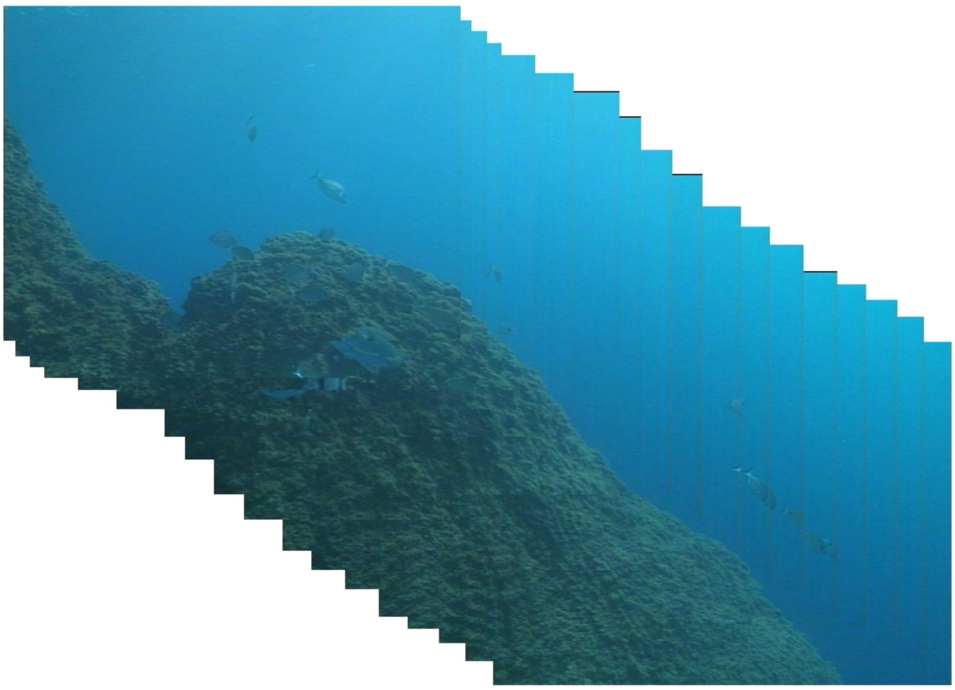

Fig. 6: Mosaic creation from a sequence of 180 images. One image every 10 is considered for actualising the mosaic