IAV2004 - PREPRINTS
5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles
Instituto Superior Técnico, Lisboa, Portugal
July 5-7, 2004

LINEAR GLOBAL MOSAICS FOR UNDERWATER SURVEYING

Nuno Gracias, João Paulo Costeira and José Santos-Victor *
Instituto Superior Técnico & Instituto de Sistemas e Robótica
Av. Rovisco Pais, 1049–001 Lisboa Codex, Portugal

Abstract. An important feature for autonomous underwater vehicles equipped with video cameras in survey missions, is the ability to quickly generate a wide area view of the sea floor. This paper presents a method for the fast creation of globally consistent video mosaics. A closed–form solution for the estimation of the global image motion is presented. It uses a least-squares criteria over a residual vector which is linear on the homography parameters. Aiming at real–time operation, a fast implementation is described using recursive least–squares, which permits the creation of globally consistent mosaics during video acquisition. The application to underwater imagery is illustrated by the creation of video mosaics capable of being used for surveying or autonomous navigation.

Key Words. Global image registration; recursive least–squares; underwater mosaicing; computer vision

1. INTRODUCTION

This paper addresses the problem of the fast creation of globally consistent mosaics. We present a simple formulation based on an affine description of the image motion. This model allows for formulating the mosaic creation problem as the minimization of the norm of a vector of residues which is a sparse linear combination of the coordinates of point sets resulting from matching several image pairs. The linear nature of the problem allows for obtaining fast solutions using least squares methods.

The methodology in this paper can be used for efficiently creating navigation maps for autonomous underwater vehicles, or in ROV–assisted human surveying of an underwater region. For computer vision applications requiring higher registration accuracy, the method is valuable in providing an initial global motion estimate. This estimate can serve as the initial value for a posterior finer registration step, involving more specific motion models and non–linear optimization.

An important feature for autonomous underwater vehicles equipped with video cameras is the ability to quickly generate a wide area view of the sea floor. Such view can easily be interpreted by a human operator on a survey mission or be used as a spatial representation for navigation. When compared with land or aerial environments, the light underwater is subject to intense attenuation and scattering. These factors severely limit the definition and range of underwater imagery. Under such conditions, video mosaicing methods are suited to creating large visual representations of the sea floor, through the registration of many close–range images.

Underwater video mapping commonly requires the registration of large sets of images of the region of interest (Gracias *et al.*, 2003; Negahdaripour and Firoozfam, 2001). Most commonly the image registration is performed by pair–wise image registration in chronological order (Gracias and Santos-Victor, 2000; Plakas and Trucco, 2000). The resulting motion estimates are then concatenated to infer the relation between any pair of images. However, even small amounts of noise in the estimation process may result in large accumulated error. This is most noticeable if the image sequence contains regions of the scene that have been captured some time before, such as loop camera trajectories.

1.1. Related work on global registration

A number of authors have tackled the problem of registration for camera loop trajectories in order to create spatially coherent mosaics (Sawhney *et al.*, 1998). Bundle adjustment techniques from the photogrammetry literature have been successfully adapted to image registering applications (McLauchlan and Jaenicke, 2000). A common feature of such approaches is the use of an obser-

[0] Email: {ngracias,jpc,jasv}@isr.ist.utl.pt. This work was (partially) supported by the FCT, the FCT Programa Operacional Sociedade de Informação (POSI) in the frame of QCA III.

vation model that impose non-linear constraints on the motion parameters (Gracias and Santos-Victor, 2001; Duffin and Barrett, 1998), thus requiring off–line minimization which is often highly time–consuming.

Recently (Unnikrishnan and Kelly, 2002) addressed the problem of efficiently distorting strip mosaics in order to close loops in a smooth way. The proposed solution has low computational complexity and is best suited for the case where the number of temporally distant overlaps is small compared to the adjacent ones.

In (Davis, 1998), a least squares solution is outlined for the global registration of images captured under no translation. The elements of pair-wise homographies are used as data in a linear system of equations for registering each image on a common reference frame. However, the issue of independent scale factors arising from the use of projective homographies is not addressed.

Garcia *et al.*(Garcia *et al.*, 2002) address the problem of estimating the position of an AUV while constructing a mosaic. The issue of looping trajectories is dealt with using a Kalman Filter with an augmented state vector. Part of our paper address the same issues, but using a recursive least squares framework, where there is no concern in explicitly estimating the position of the camera with respect to the mosaic, nor the need of a dynamic model for the motion of the vehicle deploying the camera.

The simple observation structure that arises from using the affine motion model for global registration has been overlooked in the mosaic creation literature. The main contribution of our paper lies on the formulation of the global registration problem in the least squares framework. This framework enables a simple and fast implementation, which we illustrate on underwater mosaics.

2. METHODOLOGY

The methodology in this paper is divided into two parts. The first addresses the registration as a batch, using a linear least squares criteria. The second describes a recursive formulation intended for real–time operation.

2.1. Least Squares Solution to the Global Mosaic

In this section we will assume that a sequence of image frames have been acquired. For each pair of overlapping images (i, j), we will assume that a number P_{ij} of point correspondences have been found. Let $x^n_{i,j} = \left(u^n_{i,j}, v^n_{i,j}\right)$ be the 2–D image coordinates of the n^{th} measured in the coordinate frame of image j, which matches point $x^n_{j,i}$, measured in the coordinate frame of image i.

We will now address the problem of global image motion estimation. As we are interested in obtaining a fast solution, we will formulate the problem as the minimization of the norm of a residual vector which is linear with the image motion parameters. The residual vector contains the differences in the coordinates of selected image points which are in correspondence and are mapped to a common reference frame.

We will assume that the image motion between frames can be adequately described by an affine model, represented by a 3×3 affine homography matrix of 6 parameters. Ideally, for the application cases where there is unconstrained camera motion (such as 3D camera translation and rotation), one would use a full projective collineation as this is the most general and accurate model for describing the mapping between two or more images obtained by projective cameras looking at a plane. However, the affine model is the most general model which allows for the residual vector to be expressed as a linear combination of the motion parameters for more than two images.

The following linear formulation assumes the presence of a common reference frame where the residuals are measured. Let $H_{Ref,i}$ be the affine homography that maps the coordinate frame of image i onto the reference frame, such that

$$H_{Ref,i} = \begin{bmatrix} h_1^{(i)} & h_2^{(i)} & h_3^{(i)} \\ h_4^{(i)} & h_5^{(i)} & h_6^{(i)} \\ 0 & 0 & 1 \end{bmatrix}.$$

Let $H_{i,j}$ be the affine homography relating frames i and j. The corresponding point residues are defined as $r^n_{i,j} = C\left(x^n_{j,i}\right) \cdot \overrightarrow{h^{(i)}} - C\left(x^n_{i,j}\right) \cdot \overrightarrow{h^{(j)}}$, where

$$C(x) = \begin{bmatrix} u & v & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & u & v & 1 \end{bmatrix}$$
$$\overrightarrow{h^{(i)}} = \begin{bmatrix} h_1^{(i)} & h_2^{(i)} & h_3^{(i)} & h_4^{(i)} & h_5^{(i)} & h_6^{(i)} \end{bmatrix}^T.$$

Given a set of N images, we assume that a set of M homographies were found by pair–wise matching, such that the whole image set is *connected*, *i.e.*, every image can be related to every other by appropriately cascading the homographies. This condition implies $M \geq N - 1$, where the case $M = N - 1$ would correspond to having each image matched with only one other image, such as the case of simple time sequential matching.

Let $\widetilde{\beta}$ be defined as the $6N \times 1$ vector containing all the stacked elements of $H_{Ref,i}$ for all images, $\widetilde{\beta} = \left[h_1^{(1)} ... h_6^{(1)} h_1^{(2)} ... h_6^{(N)} \right]^T$. By combining the equations of the point residues for all points, a linear system of equations can be written in the form

$$R = \widetilde{X} \cdot \widetilde{\beta} \tag{2.1}$$

where R is a vector of stacked point residues and $\widetilde{X}$ is a sparse matrix obtained from the coordinates of the point matches. For a total of P matched points, $\widetilde{X}$ is sized $2P \times 6N$ and has a maximum of $6P$ non-zero elements.

The sough solution will be obtained by minimizing the norm of the residues vector. However, in order to find a unique solution we must establish the common reference frame where the point coordinate differences are measured. A simple way to do this, is to select one of the image frames as the common mosaic reference frame. Without loss of generality, we will select the first image frame as the reference frame, which is expressed as $\overrightarrow{h^{(1)}} = \left[\begin{array}{cccccc} 1 & 0 & 0 & 0 & 1 & 0 \end{array} \right]^T$.

An unconstrained system of linear equations can formed from eq. 2.1 by excluding $\overrightarrow{h^{(1)}}$ from $\widehat{\beta}$. Let β be defined as $\beta = \left[h_1^{(2)} ... h_6^{(2)} h_1^{(3)} ... h_6^{(N)} \right]^T$. Equation 2.1 can be expressed as

$$R = \left[\begin{array}{cc} \widetilde{X_1} & X \end{array} \right] \cdot \left[\begin{array}{c} \overrightarrow{h^{(1)}} \\ \beta \end{array} \right] = X \cdot \beta - Y$$

where $\widetilde{X_1}$ and X are matrices of sizes a $2P \times 6$ and $2P \times 6\,(N-1)$ respectively, and $Y = -\widetilde{X_1} \cdot \overrightarrow{h^{(1)}}$.

Using the L_2 norm, the global mosaicing problem can be stated as the classic unconstrained least squares problem of finding the estimate $\widehat{\beta}$ such that $\widehat{\beta} = \arg\min_{\beta} \|X \cdot \beta - Y\|_2$, for which the closed form solution is $\widehat{\beta} = \left(X^T X \right)^{-1} \cdot Y$. However the computation of $X^T X$ should be avoided as it can lead to a large condition number and thus limit the accuracy of the solution. In this paper we have used a method based on the QR decomposition of X (Press *et al.*, 1988).

The simplicity of the least squares formulation and the sparse structure of the matrix X allow for a fast solution to the global mosaicing problem. This motivates a recursive least squares formulation, suited for real–time applications, that will be presented in the following section.

2.2. Recursive Least Squares

We will now assume that we have a image stream and that we are able to match each new incoming image with (a least) the previous incoming image.

The recursive formulation is based on two distinct estimate updates, namely an *observation update* and an *order update*. The observation update corresponds to the inclusion of new data arising from a match between a pair of previously acquired images, whereas the order update corresponds to the inclusion of a new image.

For the order update we assume that the new image is only matched with the previous one. This assumption is validated by the fact that, in practical applications, we have large superposition between time–adjacent frames and thus the matching of a new image with the previous has a high probability of being successful. Furthermore, we will take advantage of the special observations structure.

The general notation for the recursive formulation is the following. Let $\widehat{\beta}_t$, X_t and Y_t be the instances at discrete time instant t,of $\widehat{\beta}$, X and Y. Let N_t be the number of images at t, so that $\widehat{\beta}_t$ is a $6\,(N_t - 1) \times 1$ vector. As before, we will consider the first image to be the reference frame.

2.2.1. Observation Update

A new observation update corresponds to the appending a $2P_{ij} \times 6\,(N_t - 1)$ matrix x_t to X_{t-1} and the corresponding $2P_{ij} \times 1$ vector y_t to Y_{t-1}. The updated estimate $\widehat{\beta}_t$ is obtained recursively from $\widehat{\beta}_{t-1}$, involving X_{t-1}, Y_{t-1}, x_t and y_t. The simplest recursive formulation involves the storage and updating of the inverse of the autocorrelation matrix $M_t^{-1} = \left(X_t^T X_t \right)^{-1}$. However we have used the square root filter which maintains a factorization of the form $M_t^{-1} = S_t S_t^T$ and compares favorably in terms of stability. Details on the formulation and implementation of this filter can be found in (Pollock, 1999).

2.2.2. Order Update

The order update involves enlarging $\widehat{\beta}_{t-1}$ to accommodate the homography parameters for the new image. The new estimate is found by solving $X_t \cdot \widehat{\beta}_t = Y_t$ where

$$X_t = \left[\begin{array}{cc} X_{t-1} & \mathbf{0} \\ A & B \end{array} \right], \widehat{\beta}_t = \left[\begin{array}{c} \widehat{\beta}_{t-1} \\ h^{(N_t)} \end{array} \right]$$
$$\text{and } Y_t = \left[\begin{array}{c} Y_{t-1} \\ y_t \end{array} \right].$$

Matrices A and B are sized $2P_{n-1,n} \times 6\,(N_t - 2)$ and $2P_{n-1,n} \times 6$ respectively. The updating of $\widehat{\beta}_t$ implies the computation of $h^{(N_t)}$ which can be obtained by solving

$$A \cdot \widehat{\beta}_{t-1} + B \cdot h^{(N_t)} = y_t.$$

As we assume that each new image is matched with the previous, A as the following structure $A = \left[\begin{array}{cc} \mathbf{0} & D \end{array}\right]$ where D is a $2P_{n-1,n} \times 6$ matrix. Using a least squares criteria, $h^{(N_t)}$ is given by

$$h^{(N_t)} = \left(B^T B\right)^{-1} B^T \left(y_t - D \cdot h^{(N_t-1)}\right)$$

where $h^{(N_t-1)}$ are the lower 6 elements of $\widehat{\beta}_{t-1}$. Note that the computation of $h^{(N_t)}$ is a fast process due to the small sizes of B and D.

We now need to update S_t in order to use the square root filter posteriorly. The relation between S_t and S_{t-1} can be compactly expressed as

$$\left(S_t S_t^T\right)^{-1} = \left[\begin{array}{cc} G & A^T B \\ B^T A & B^T B \end{array}\right]$$

where $G = \left(S_{t-1} S_{t-1}^T\right)^{-1} + A^T A$. Note that for a large number of images the matrix G will be much larger than $B^T B$.

Taking into consideration the execution speed requirements, we are interested in computing $S_t S_t^T$ without requiring the explicit inversion of G. Using the formulas of inversion by partition $S_t S_t^T$ can be computed as

$$S_t S_t^T = \left[\begin{array}{cc} P & Q \\ Q^T & T \end{array}\right]$$

where

$$\begin{array}{l} T = \left(B^T B - B^T A G^{-1} A^T B\right)^{-1} \\ Q = -\left(G^{-1} A^T B\right) \cdot T \\ P = G^{-1} - Q Q^T G^{-1} \ . \end{array}$$

Having G^{-1}, the above expressions require few matrix additions and multiplications, and the inversion of a $2P_{n-1,n} \times 2P_{n-1,n}$ matrix. The inverse of G can be efficiently computed using the Woodbury formula,

$$\begin{array}{l} G^{-1} = \left(E^{-1} + A^T A\right)^{-1} = \\ E - \left[E \cdot A^T \cdot \left(I + A \cdot E \cdot A^T\right)^{-1} \cdot A \cdot E\right] \end{array}$$

where $E = S_{t-1} S_{t-1}^T$ and I is the $2P_{n-1,n} \times 2P_{n-1,n}$ identity matrix. Again, the above formula implies the inversion of a $2P_{n-1,n} \times 2P_{n-1,n}$ matrix. Finally, S_t is recovered from $S_t S_t^T$ using a Choleski factorization algorithm.

3. IMPLEMENTATION

This section details the algorithms we used to validate the approach and discusses some implementation details that influence the performance. All benchmarks refers to a 1.6GHz processor and acquired images of 180 × 135 pixels.

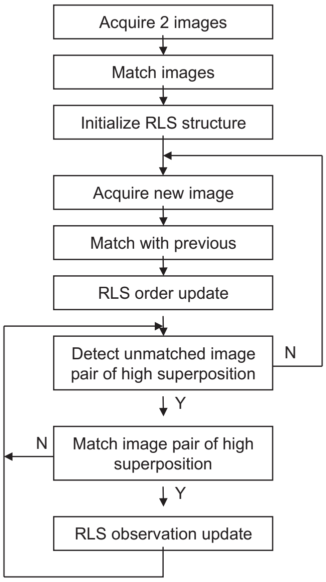

Fig. 3.1. Step sequence for the recursive construction of the mosaic.

3.1. Pair wise matching

An essential building block for the methods in this paper is the algorithm for finding point correspondences between two images of the same planar scene. The algorithm is summarized in the following. Further details can be found in (Gracias, 2003).

A set of point features corresponding to textured areas, is extracted from one of the images. For each feature (defined as a small square image patch centered at the detected corner location), a prospective match is found in the other image, using normalized cross-correlation. A robust estimation technique is used to remove outliers using a Least Median of Squares criterion, and random sampling. Typical total execution time is 1 second, for 50 inliers selected out of 70 matched features.

3.2. Recursive Mosaic Algorithm

An algorithm for the recursive construction of mosaics was implemented. This algorithm combines the pair-wise image matching, superposition detection and recursive least squares updates, to create a globally consistent mosaic on–line. The overall algorithmic flow is presented in Figure 3.1.

The recursive procedure requires the initialization the data structures it maintains. The initial values of $\hat{\beta}_0$, X_0, Y_0 and S_0 are obtained from the point matches of two initial frames. Next, the algorithm contains two nested cycles. The outer cycle corresponds to the inclusion of a new image while the inner cycle exploits the superposition between previously acquired images.

A new image is matched over the last one. The resulting point matches are used to update the order of the RLS filter, as described in Section 2.2.2. As this image may also overlap other images, we search for superposition between non–consecutive images. If large overlap is found between an unmatched image pair, the pair–wise matching is attempted. If it succeeds, then $\hat{\beta}$ is updated using the square root filter. This cycle of superposition detection, matching and RLS update is performed until no unmatched overlapping image pairs are found or all attempted image matching fails. Then a new image is processed.

We measure the amount of superposition between any pair of images by composing the corresponding inter–image homography, using the current $\hat{\beta}$.

3.3. Implementation Considerations

Both batch and recursive least squares methods can straightforwardly be adapted more restricted models of image motion. For the case of underwater or aerial surveying, it is often preferable to use the 4 d.o.f. similarity, which is suited for setups where the image plane is approximately parallel to the scene (Gracias, 2003). This motion model was used in some of the experiments. The image to reference frame homography is defined as

$$H_{Ref,i} = \begin{bmatrix} h_1^{(i)} & -h_2^{(i)} & h_3^{(i)} \\ h_2^{(i)} & h_1^{(i)} & h_4^{(i)} \\ 0 & 0 & 1 \end{bmatrix},$$

and $\overrightarrow{h^{(i)}} = \begin{bmatrix} h_1^{(i)} & h_2^{(i)} & h_3^{(i)} & h_4^{(i)} \end{bmatrix}^T$. All other matrices and vectors are sized accordingly.

To promote the execution speed, the number of point coordinates used in the global estimation methods was reduced to the minimum required for the motion model, namely 3 points for the affine model and 2 for the similarity. Note that the pairwise image matching is performed with a large set of points, and that the resulting homography is computed from a large set of inliers. The homography is used to compute 2 (or 3) virtual points, for the global estimation.

4. RESULTS

The following results were obtained using underwater video sequences captured from a submersible. The first sequence contains 85 images that were acquired while the camera was undergoing a 3 loop trajectory. The average superposition between time consecutive frames is 55%. Figure 4.1 shows the result of from simple sequential image matching. Several sources of error, such as non–planar scene, limited matching resolution and affine camera model, lead to the error accumulation which is visible in the repetitive patterns corresponding to the same ground features. The mosaic of Figure 4.2 was created using batch least squares with the affine motion model and 309 pairs of matched images.

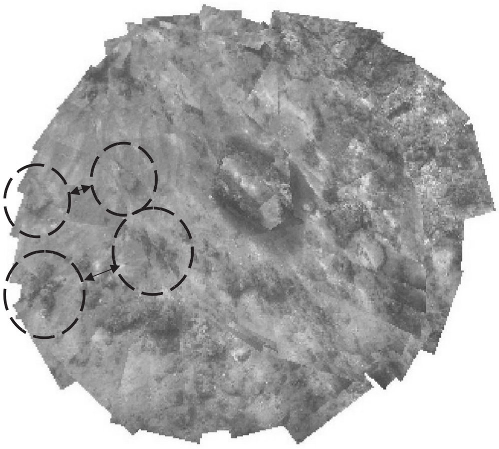

Fig. 4.1. Mosaic created from sequential image matching. The effect of the accumulated error is visible on the marked regions, corresponding to the same features on the sea floor.

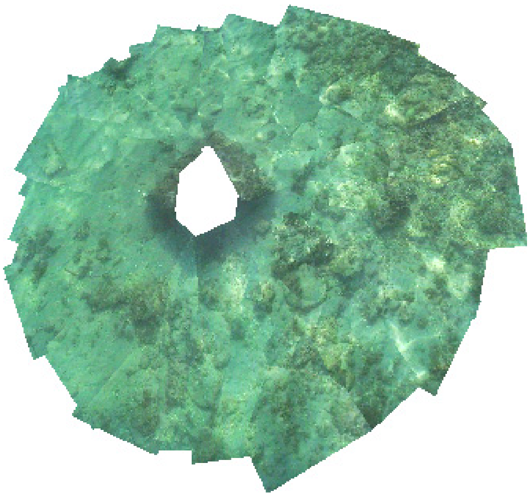

Fig. 4.2. Global mosaic using batch least squares.

A second sequence of 129 image frames, selected from a larger set of 6 minutes of video, was used to create the mosaic of Figure 4.3, using the recursive mosaic algorithm. Upon completion, 272 pairs of images were matched.

For comparison, Table 4.1 presents the optimization time required to obtain a global motion estimate using batch (BLS) and non–linear least squares (NLLS). The NLLS method is described

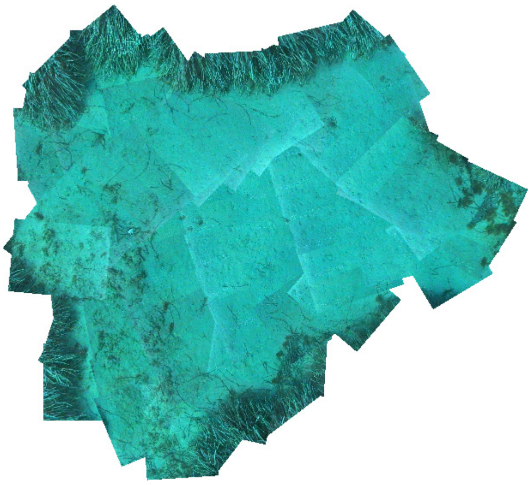

Fig. 4.3. Global mosaic using recursive least squares.

Sequence	model	BLS	NLLS
First	Affine	0.16	13
Second	Similarity	0.14	20

Table 4.1 Execution time (in seconds) for batch (BLS) and non–linear least squares (NLLS).

in (Gracias, 2003), where it was used for topology estimation. Although much slower, the NLLS presents the advantage of coping with more specific non–linear motion models, such as the constant scale similarity.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a least squares approach for the creation of globally consistent mosaics. The approach allows for a linear formulation using point matches between pairs of images, and the fast estimation of the image motion. Both batch and recursive implementations were detailed. Due to the low computational demand, an important advantage of the recursive implementation is the possibility of performing the image registration, loop detection and trajectory correction in an integrated fashion. For underwater vision applications, this methodology allows for the creation of wide views of the sea floor during image acquisition, thus being of benefit in human surveying mission or in map building for autonomous navigation.

Future work includes the extension of the method to deal with geographic data, such as sensor readings during acquisition, and world points of known location. We are also investigating the use of non–linear image motion models without resorting to bundle adjustment techniques.

6. REFERENCES

Davis, J. (1998). Mosaics of scenes with moving objects. In: *Proc. of the Conference on Computer Vision and Pattern Recognition*. Santa Barbara, CA, USA.

Duffin, K. and W. Barrett (1998). Globally optimal image mosaics. In: *Graphics Interface*. pp. 217–222.

Garcia, R., J. Puig, P. Ridao and X. Cufi (2002). Augmented state Kalman filtering for AUV navigation. In: *Proc. Int. Conf. on Robotics and Automation*. Washington DC, USA. pp. 4010–4015.

Gracias, N. (2003). Mosaic–based Visual Navigation for Autonomous Underwater Vehicles. PhD thesis. Instituto Superior Técnico. Lisbon, Portugal.

Gracias, N. and J. Santos-Victor (2000). Underwater video mosaics as visual navigation maps. *Computer Vision and Image Understanding* **79**(1), 66–91.

Gracias, N. and J. Santos-Victor (2001). Underwater mosaicing and trajectory reconstruction using global alignment. In: *Proc. of the Oceans 2001 Conference*. Honolulu, Hawaii, U.S.A.. pp. 2557–2563.

Gracias, N., S. Zwaan, A. Bernardino and J. Santos-Victor (2003). Mosaic based Navigation for Autonomous Underwater Vehicles. *Journal of Oceanic Engineering*.

McLauchlan, P. and A. Jaenicke (2000). Image mosaicing using sequential bundle adjustment. In: *Proc. of the British Machine Vision Conference BMVC2000*. Bristol, U.K.

Negahdaripour, S. and P. Firoozfam (2001). Positioning and image mosaicing of long image sequences; Comparison of selected methods. In: *Proc. of the IEEE Oceans 2001 Conference*. Honolulu, Hawai, USA.

Plakas, C. and E. Trucco (2000). Developing a real-time robust video tracker. In: *Proc. of the IEEE Oceans 2002 Conference*. Providence, Rhode Island, USA.

Pollock, D. (1999). *A Handbook of Time–Series Analysis, Signal Processing and Dynamics*. Academic Press.

Press, W., S. Teukolsky, W. Vetterling and B. Flannery (1988). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press.

Sawhney, H., S. Hsu and R. Kumar (1998). Robust video mosaicing through topology inference and local to global alignment. In: *Proc. European Conf. on Computer Vision*. Freiburg, Germany.

Unnikrishnan, R. and A. Kelly (2002). A constrained optimization approach to globally consistent mapping. In: *Proc. Int. Conf. on Intell. Robots and Systems*. Lausanne, Switzerland.