

Correction of Shading Effects in Vision-Based UUV Localization

R. Garcia, X. Cufi, M. Carreras and P. Ridao
Computer Vision and Robotics Group
University of Girona
17071 Girona, Spain

Abstract—This paper describes the improvements achieved in our mosaicking system to assist Unmanned Underwater Vehicle navigation. A major advance has been attained in the processing of images of the ocean floor when light absorption effects are evident. Due to the absorption of natural light, underwater vehicles often require artificial light sources attached to them to provide the adequate illumination for processing underwater images. Unfortunately, these flashlights tend to illuminate the scene in a nonuniform fashion. In this paper a technique to correct non-uniform lighting is proposed. The acquired frames are compensated through a point-by-point division of the image by an estimation of the illumination field. Then, the gray-levels of the obtained image are remapped to enhance image contrast. Experiments with real images are presented.

I. INTRODUCTION

When *Unmanned Underwater Vehicles* (UUVs) perform missions near the ocean floor, optical sensors can be used to improve local navigation [1,2]. A mosaicking system can be used to process the images provided by a down-looking camera carried by the vehicle. In this way, the acquired images can be used to refine the estimation of the vehicle's position when it is navigating near the ocean floor [3].

However, light gets attenuated and scattered in the underwater medium, disappearing from the image-forming process [4,5]. This process of absorption of the medium generates serious problems in underwater images, such as blurring of image features, limited range, clutter and lack of structure in the regions of interest. Often, as depth increases, natural light is not sufficient for imaging the sea floor. For this reason, a light source is normally attached to the submersible providing the necessary lighting. Artificial light sources tend to illuminate the scene in a nonuniform fashion, producing a bright spot in the center of the image with a poorly illuminated area surrounding it. Therefore, application of standard computer vision techniques to underwater imaging requires dealing first with these added problems.

In this paper we propose a revision of our mosaicking algorithm [6], adapting it for dealing with changes in the brightness of the imagery as the vehicle moves. The remaining of the paper is organized as follows. First, section II gives a brief description of our mosaicking algorithm. Then, section III proposes an image enhancement technique to compensate nonuniform lighting effects. Next, some experiments performed on real data acquired by the under-

water vehicle URIS are shown in section IV; and, finally, the conclusions and future work close the paper.

II. MOTION ESTIMATION THROUGH MOSAICKING

Our mosaicking system is intended to work on either URIS or GARBI, the two UUVs developed at the Computer Vision and Robotics Group of the University of Girona. Depending on the sensors available in every vehicle, the mosaicking system may take advantage of additional information provided by the on-board sensors (*e.g.* altimeter sonar, compass, Doppler Velocity Log, Inertial Navigation System, inclinometer, etc.). A Sensor Fusion module (SF) integrates an estimation of the motion of the vehicle. Detailed description of this module is out of the scope of this paper, since it depends on which vehicle we consider, and which on-board sensors are available. In the best case, the SF module would provide incremental measurements on position and orientation $(x, y, z, \phi, \theta, \psi)$. However, since we plan our mosaicking system to work under different conditions, this estimation may not be available, or may be inaccurate (*e.g.*, the robot may only carry a gyrocompass, which means that only *yaw* will be available). Therefore, we can assume without loss of generality that the SF module provides the necessary information to compute a planar homography H_c^e with 8 degrees of freedom, which encodes any available information (converting pitch and roll into rough estimations of perspective deformation) [7].

The mosaicking system is divided into two main blocks, namely: *mosaic supervisor* and *mosaic engine*. The *mosaic supervisor* keeps the state of the system and takes decisions according to this state. It takes into account the estimations of the SF module (which depend on the vehicle and its configuration), analyses how the vehicle is moving and generates the pertinent orders for the *mosaic engine*. The *supervisor* module is responsible of the mosaic data structure, that is, updating the mosaic image (I_m), according to the acquired images. It provides the *engine* with the images which will be used to estimate the motion of the vehicle. One of these images is the current frame acquired by the camera (I_c). The second one is a reference image (I_r) which has been extracted from the photo-mosaic at the present location. Along with these images the engine provides an initial estimate of the apparent motion between them. This initial estimate of the apparent motion is mate-

rialized in the form of the “a priori” 3×3 estimation matrix ${}^r\mathbf{H}'_c$. This matrix can be computed from the data provided by the SF module, or, if on-board sensor information is not available, the *supervisor* can generate ${}^r\mathbf{H}'_c$ based uniquely on the vision system, as will be described below. The output of the mosaic *engine* is another matrix, ${}^r\mathbf{H}_c$, which provides a refinement of the initial motion estimation and its associated uncertainty. The *engine* is controlled by the mosaic *supervisor*. Therefore, it is only executed when the *supervisor* requires an iteration of the *engine*, providing a matrix ${}^r\mathbf{H}_c$ which describes the planar motion between images I_c and I_r .

The operations performed by the *supervisor* are the following. First, image I_c is acquired by the camera, and the geometric distortion caused by the lens and the camera housing is corrected. The next step consists of selecting image I_r to be passed jointly with image I_c to the engine. At time instant k , the supervisor uses matrix ${}^r\mathbf{H}'_c$ to check if the overlapping between the previous reference image $I_r(k-1)$ and the current one $I_c(k)$ is below a desired threshold. If this is the case, it will select a new reference image $I_r(k)$. The new reference image will be extracted from the mosaic image $I_m(k-1)$ at the same position and orientation as that of the last image processed by the system, *i.e.*, the current image at the previous time instant $I_c(k-1)$. If the overlap between images $I_c(k)$ and $I_r(k-1)$ is bigger than the threshold, the reference image will not change, *i.e.* $I_r(k) = I_r(k-1)$. Finally, images $I_r(k)$ and $I_c(k)$, and matrix ${}^r\mathbf{H}'_c$ are passed to the mosaic engine, and it is told to execute.

The *engine* then begins its execution by detecting interest points in image I_c . The goal of our interest point detector is to find stable features in the image, *i.e.* scene features which can be reliably found when the camera moves from one location to another and lighting conditions of the scene change somewhat. The strategy which has been used to detect these interest points is based on the use of a corner detector [8].

Once the interest points have been detected in image I_c , the next step consists of finding their correspondences in the reference image I_r . Before searching for correspondences, both images are smoothed with a 3×3 Gaussian mask. Given an interest point ${}^c\mathbf{m}$ in image I_c , instead of considering the point as an individual feature, an $n \times n$ region $R({}^c\mathbf{m})$ centered at this point is selected. Then, the *engine* aims to find a point ${}^r\mathbf{m}$ in reference image I_r , surrounded by a $n \times n$ area $R({}^r\mathbf{m})$, which presents a high degree of similarity to $R({}^c\mathbf{m})$. This “similarity” is computed as a correlation function [9]. In order to reduce the computational cost, the considered regions, $R({}^c\mathbf{m})$ and $R({}^r\mathbf{m})$, are subsampled by a factor q , reducing the processed pixels from $n \times n$ to $m \times m$, where $m = ((n-1)/q) + 1$, as shown in equation (1).

$$\text{corr}({}^c\mathbf{m}, {}^r\mathbf{m}) = \frac{\sum_{i=-a}^a \sum_{j=-a}^a (I_c(x+i \cdot q, y+j \cdot q) - \overline{I_c(x,y)}) \cdot (I_r(u+i \cdot q, v+j \cdot q) - \overline{I_r(u,v)})}{(2a+1)^2 \sqrt{\sigma^2(I_c) \cdot \sigma^2(I_r)}}, \quad (1)$$

in which $a = (n-1)/2q$, I_c and I_r are the current and reference images for which the motion is to be computed, $\sigma^2(I)$ is the variance of the image computed in the correlation window (see equation (2)); and $\overline{I(x,y)}$ is the average of the correlation window in the image as shown in equation (3).

$$\sigma^2(I) = \frac{\sum_{i=-a}^a \sum_{j=-a}^a I(x+i \cdot q, y+j \cdot q)^2}{(2a+1)^2} - \overline{I(x,y)}^2, \quad (2)$$

$$\overline{I(x,y)} = \frac{\sum_{i=-a}^a \sum_{j=-a}^a I(x+i \cdot q, y+j \cdot q)}{(2a+1)^2}. \quad (3)$$

Equation (1) is normalized by subtracting the mean and dividing by a factor which takes into account the dispersion of the gray levels in the considered regions. For this reason, this measurement of correlation is very adequate for underwater imaging, where lighting inhomogeneities are frequent. Unfortunately, although equation (1) produces good results in absence of rotations, its reliability decreases as images I_c and I_r present a higher rotational component. For this reason the reference image extracted from the mosaic has the orientation of the last image acquired by the system $I_c(k-1)$. The high frame rate guarantees the difference in orientation and scaling between two consecutive images will be very small. On the other hand, it has been proved that the accuracy of subsampling the correlation window is practically the same as using the full window [10]. This is due to the strong correlation in the gray level of neighboring pixels, producing smooth intensity variations, especially after a low-pass filtering.

According to equation (1), given an interest point ${}^c\mathbf{m}$, its correspondence ${}^r\mathbf{m}$ should be the point which has obtained the highest correlation score. Unfortunately, experimental work with underwater images has proved that in some cases the true correspondence is not the one with the highest correlation score, although its score is normally quite high. For this reason we also use *texture* as a similarity measure to solve the correspondence problem (see [11] for further details).

Once the pairs of corresponding points have been detected, those points which do not describe the dominant

motion between both images are deleted through LMedS, a robust outlier detector [12]. Next, the apparent motion between current $I_c(k)$ and reference $I_r(k)$ images is computed from the remaining pairs of points, giving rise to a 3×3 matrix ${}^r\mathbf{H}_c$. When the *engine* completes its execution, it gives back the control to the mosaic *supervisor*. The mosaic *supervisor* then decides whether image $I_c(k)$ should be merged with the composite mosaic image or, on the contrary, should only be taken into account to update the positioning estimation of the vehicle. Finally, the mosaic image $I_m(k)$ is updated and 3D positioning is obtained through triangulation by taking into account the camera focal length and a sonar altimeter (if available) [13].

In some cases the available on-board sensors may not be enough to provide “a priori” estimation matrix ${}^r\mathbf{H}'_c$. In this situation an additional execution of the engine is required. From the previous iteration, the supervisor knows the planar motion between the last current image $I_c(k-1)$ and the reference one, being ${}^r\mathbf{H}_c(k-1)$ the homography provided by the mosaic engine at the previous instant. Then, the controller asks the engine to compute the motion between the last two acquired images: $I_c(k-1)$ and $I_c(k)$. The resulting homography is multiplied by ${}^r\mathbf{H}_c(k-1)$, and in this way “a priori” estimation matrix ${}^r\mathbf{H}'_c(k)$ is obtained. Once matrix ${}^r\mathbf{H}'_c(k)$ has been obtained, the controller provides the engine with this matrix and the current and reference images, with which the engine performs a second iteration.

III. CORRECTION OF LIGHTING EFFECTS

The previous section has given a brief description of our mosaicking system. Although the mosaic engine is able to register two images presenting slight variations in their radiosity, better results can be obtained if the images undergo a process of radiometric rectification prior to the motion estimation phase. In this section, a strategy for correcting non-uniform lighting is described. This strategy exploits the illumination-reflectance model by considering the image as a function of the product of the reflectance properties of the imaged objects, and the illumination field present in the scene [14], as shown in equation (4).

$$f(x, y) = i(x, y) \cdot r(x, y), \quad (4)$$

where $f(x, y)$ is the image sensed by the camera, $r(x, y)$ is the reflectance function (or ideal image under absence of shading) and $i(x, y)$ represents the illumination multiplicative factor. Depending upon the camera characteristics, it may also contribute two more terms: (i) a gain component $g(x, y)$ and (ii) an offset term $o(x, y)$:

$$f(x, y) = g(x, y) \cdot i(x, y) \cdot r(x, y) + o(x, y), \quad (5)$$

Depending on the camera characteristics, gain and offset terms may be a function of (x, y) , or may also be two constants g and o , instead of $g(x, y)$ and $o(x, y)$. In any case, arranging equation (5), the sensed image can be expressed as a reflectance function adjusted by a multiplicative $c_m(x, y)$ and an additive $c_a(x, y)$ shading component:

$$f(x, y) = c_m(x, y) \cdot r(x, y) + c_a(x, y), \quad (6)$$

Normally, the multiplicative factor $c_m(x, y)$ due to light sources carried by the vehicle and camera sensitivity can be modeled as a smooth function. In order to model this non-uniform illumination, a Gaussian-smoothed version of the image acquired by the camera $f(x, y)$ is proposed. The smoothed image $f_s(x, y)$ is intended to be an estimate of how much the illumination field (and camera sensitivity) affects every pixel of the image. To obtain this effect, the smoothing has to be large compared to the size of the features in the image. Therefore, the acquired image can be corrected by a point-by-point division by the smoothed image, giving rise to an estimate of ideal image $\tilde{r}(x, y)$ can be obtained through:

$$\tilde{r}(x, y) = \frac{f(x, y)}{f_s(x, y)} \cdot \delta, \quad (7)$$

where δ is a normalization constant which restores the overall image luminance. Here we have deliberately ignored offset term $c_a(x, y)$ which could come from non-uniform camera sensitivity since in an underwater environment its influence is quite small with respect to $i(x, y)$, at least for a standard camera.

Next, the contrast of the resulting image is emphasized through equation (8), giving rise to $\tilde{r}'(x, y)$, an “equalized” version of $\tilde{r}(x, y)$.

$$\tilde{r}'(x, y) = \left[\tilde{r}(x, y) - \min(\tilde{r}(x, y)) \right] \frac{\tau_H - \tau_L}{\max(\tilde{r}(x, y)) - \min(\tilde{r}(x, y))} + \tau_L, \quad (8)$$

where τ_H and τ_L , are, respectively, the maximum and minimum desired values of $\tilde{r}'(x, y)$; $\max(\tilde{r}(x, y))$ represents the brightest gray level of image $\tilde{r}(x, y)$ which is smaller than 1.5 times the third quartile of the histogram of $\tilde{r}(x, y)$; and a similar approach is applied to $\min(\tilde{r}(x, y))$ with the first quartile. In this way the proposed methodology increases its robustness with respect to noise.

Moreover, in our implementation the smoothed image is not computed for every image of the sequence. It is only computed from a set of consecutive frames. The result is averaged, and then a 2D-Gaussian function is adjusted to the average image. In this way, it is used in equation (7) for every new image, thus saving computational effort. However, this simplification is only valid if the vehicle keeps constant its distance to the ocean floor (known as *altitude*). Fortunately, this is almost always the case when the vehicle is mosaicking the ocean floor.

IV. EXPERIMENTAL RESULTS

Some experiments have been carried out to test the performance of the lighting-correction strategy as a preliminary step for the construction of photo-mosaics. The images have been acquired by URIS in shallow waters of Costa Brava at sundown, to avoid the influence of the sun rays, thus simulating Deep Ocean operation. In these images, the vehicle carries its own light producing a serious nonuniform illumination effect in the form of a bright spot close to the center of the image. Although the mosaicking system can run in real time while the vehicle is performing its mission, the sequence of images considered in this paper has been acquired and stored to disk to be processed offline. In this way, the different approaches can be tested exactly on the same data. The considered sequence is formed by a flat area of the sea floor with sand and small rocks. However, since the underwater terrain is not perfectly flat, the motion of the vehicle, and therefore, motion of the light source, induces small shadows in the scene. These shadows move in opposite direction of the light source, which introduces an additional difficulty in the motion estimation process.

Fig. 1(left) shows two sample images of the sequence. Their size is 384×288 pixels. Smoothing these images with a large Gaussian kernel (of size 51×51) gives rise to the images displayed in Fig. 1(right). The size of the smoothed images has been kept constant. This explains the gray border around the filtered images. In these images the ocean floor presents a flat relief, with small changes in range with respect to the camera and the light source. In this situation, it is not

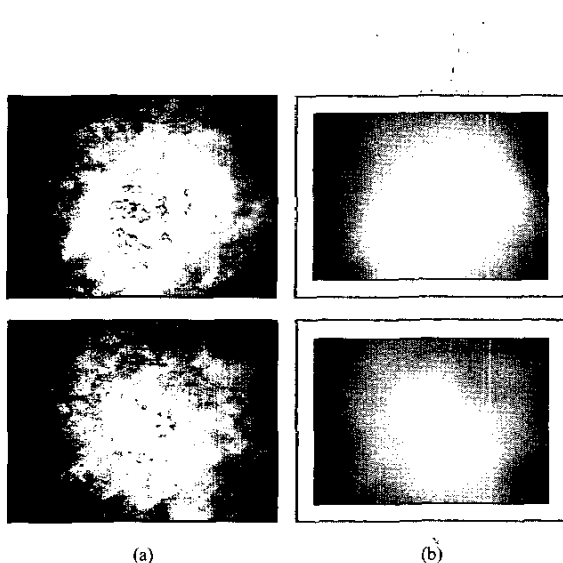


Fig. 1. Estimation of the illumination field in a sequence of images presenting nonuniform illumination. (a) Original images, (b) Result of low-pass filtering with a Gaussian kernel of size 51×51 pixels.

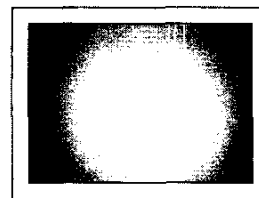


Fig. 2. Resulting image after averaging several smoothed images.

necessary to compute $f_s(x, y)$ for every acquired image, but it is computed only for the first few images of the sequence. Then, these smoothed images are averaged, and an estimation of the illumination field is obtained, as shown in Fig. 2.

Fig. 3 shows the result of correcting the illumination effects of the two images illustrated in Fig. 1(left), following the strategy described in section III. For both frames, the image of Fig. 2 has been taken as smoothed image $f_s(x, y)$. From Fig. 3 we observed that the resulting enhanced images compensate the nonuniform lighting in quite an efficient way.

Finally, the sequence of images has been used to construct a mosaic, while estimating the vehicle's trajectory. Two mosaics are shown in Fig. 4. The first one has been constructed from the original images, without correction of the illumination artifacts. The last mosaic, shown in Fig. 4(b), is the result of processing the original images with the technique described in section III, and then applying the mosaicking algorithm to the enhanced images. In both cases, no additional information from any on-board sensors has been used, in order to enable comparison between both strategies. It can be seen from Fig. 4 that the results are quite acceptable in both cases. Obviously, the mosaic with lighting correction presents a more uniform and clear appearance. Unfortunately, it is not possible to obtain an error measurement of the trajectory of the vehicle when the experiments are run in open sea, since the true trajectory is not known. However, from a qualitative (and subjective)

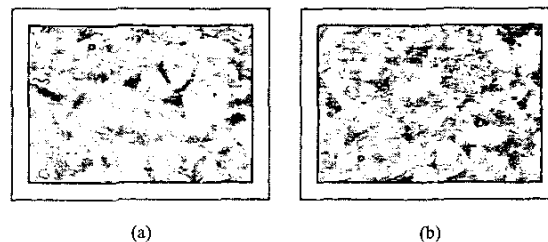


Fig. 3. Correction of non-uniform lighting. Individual frames of Fig. 1(left) after being enhanced through the strategy proposed in section III, using the Gaussian function illustrated in Fig. 2.

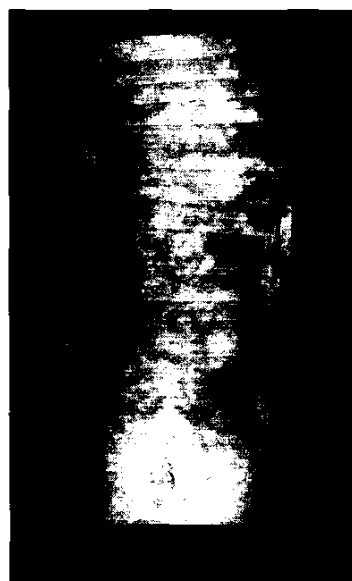
analysis, small misalignments can be perceived in some areas of the first mosaic. These alignment errors are not perceptible in the second mosaic. Moreover, comparing both images it can be observed that the first third of the mosaic of Fig. 4(a) is a bit stretched, while the last part of the mosaic has been shrunk by a small factor. In any case, this experiment proves that the performance of the textural-aided correlation is quite good in the presence of uniform differences in the illumination field, while it suffers from a small degradation when the variations in the image irradiance are more pronounced. These sharp changes occur in the visible seams between images every time a new reference image is extracted from the mosaic image (see Fig. 5). These seams do not appear in the mosaic with corrected illumination.

V. CONCLUSIONS AND FUTURE WORK

A new methodology to construct photo-mosaics of the ocean floor has been presented, achieving a number of improvements with respect to our previous work [6]. First, detection of correspondences between images provides better results by considering new texture operators [11]. Second, the propagation of drift errors as the mosaic increases its size has been reduced. This is achieved by exploiting the reference image technique, instead of processing every pair of consecutive images. Drift is kept to a minimum for several consecutive images which share the same reference image. Finally, lighting inhomogeneities can be removed from the mosaic, obtaining a mosaic without visible seams between images. This is important if the reference image has to be extracted from the mosaic image as the system evolves, avoiding the processing of images like the illustrated by Fig. 5(b).

We have seen that although correcting nonuniform illumination before the images are processed leads to more accurate trajectory estimations, the result is not so different, just a small stretching or shrinking of the estimated vehicle path. This is due to the good performance of correlation and texture analysis, which provide good results under smooth differences in the illumination field between the two images which are being processed. However, if the illumination field changes abruptly, as in the mosaic of Fig. 4 (a), this nonlinearity deteriorates the accuracy of the similarity measure. For this reason removing lighting inhomogeneities is a good option.

Future work is being focused in integrating the methodology described in [15], which was tested in simulation data, with the whole mosaicking system. In this way trajectory estimation will be improved when the vehicle re-visits areas which have been already mosaicked.

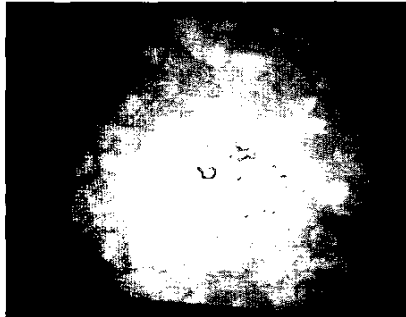


(a)



(b)

Fig. 4. (a) Mosaic constructed from the original images, without correction of the illumination artifacts. (b) Resulting mosaic obtained from the images enhanced with the technique described in section III.



(a)



(b)

Fig. 5. (a) Current image, after being rotated and scaled by the mosaic supervisor to match the characteristics of the reference image. (b) Reference image extracted from the mosaic. Better results are obtained when the reference image is extracted from a seam-less mosaic.

VI. ACKNOWLEDGMENTS

This work was partially supported by the Spanish Government (CICYT) under Grant DPI2001-2311-C03-01.

VII. REFERENCES

- [1] S. Negahdaripour, X. Xu and L. Lin, "Direct estimation of motion from sea floor images for automatic station-keeping of submersible platforms", *IEEE Journal of Oceanic Engineering*, vol. 24, no. 3, pp. 370–382, 1999.
- [2] E. Trucco, A. Doull, F. Odone, A. Fusiello and D.M. Lane, "Dynamic Video Mosaics and Augmented Reality for Subsea Inspection and Monitoring," in *Proceedings of the Oceanology International Conference*, pp. 297–306, 2000.
- [3] N. Gracias and J. Santos-Victor, "Underwater Video Mosaics as Visual Navigation Maps," *Computer Vision and Image Understanding*, vol. 79, no. 1, pp. 66–91, 2000.
- [4] C. J. Funk, S.B. Bryant and P.J. Beckman Jr., "Handbook of underwater imaging system design," Ocean Technology Department, Naval Undersea Center, 1972.
- [5] S. Negahdaripour and C.H. Yu, "On shape and range recovery from image shading for underwater applications," in J. Yuh (ed.), *Underwater Robotic Vehicles: design and control*, ch. 8, pp. 221–250, TSI Press, 1995.
- [6] R. Garcia, J. Batlle, X. Cufi and J. Amat, "Positioning an Underwater Vehicle through Image Mosaicking," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, vol. 3, pp. 2779–2784, 2001.
- [7] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge University Press, 2000.
- [8] C.G. Harris and M.J. Stephens, "A combined corner and edge detector," in *Proceedings of the Fourth Alvey Vision Conference*, Manchester, U.K., pp. 147–151, 1988.
- [9] Z. Zhang, R. Deriche, O. Faugeras, Q.T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry", INRIA RR-2273, 1994.
- [10] A. Giachetti, "Matching techniques to compute image motion," *Image and Vision Computing*, vol. 18, no. 3, pp. 247–260, 2000.
- [11] R. Garcia, X. Cufi and J. Batlle, "Detection of Matchings in a Sequence of Underwater Images through Texture Analysis," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 361–364, Thessaloniki, Greece, 2001.
- [12] P. Rousseeuw and A. Leroy, "Robust Regression and Outlier Detection", John Wiley & Sons, New York, 1987.
- [13] R. Garcia, X. Cufi and M. Carreras, "Estimating the motion of an underwater robot from a monocular image sequence," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 1682–1687, Maui, Hawaii, 2001.
- [14] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Addison-Wesley, Reading, MA, 1992.
- [15] R. Garcia, J. Puig, P. Ridao and X. Cufi, "Augmented State Kalman Filtering for AUV Navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4010–4015, Washington D.C., 2002.