Machine Vision and Applications (2002) 13: 129–140

Machine Vision and Applications

A vision system for an underwater cable tracker

Alberto Ortiz[1], **Miquel Simó**[2], **Gabriel Oliver**[3]

[1] Mathematics and Computer Science Department, University of the Balearic Islands, 07071 Palma de Mallorca, Spain; e-mail: alberto.ortiz@clust.uib.es
[2] Mathematics and Computer Science Department, University of the Balearic Islands, 07071 Palma de Mallorca, Spain
[3] Mathematics and Computer Science Department, University of the Balearic Islands, 07071 Palma de Mallorca, Spain; e-mail: goliver@clust.uib.es

Accepted: 13 August 2001

Abstract. Nowadays, the surveillance and inspection of underwater installations, such as power and telecommunication cables and pipelines, is carried out by operators that, being on the surface, drive a remotely operated vehicle (ROV) with cameras mounted over it. This is a tedious and high time-consuming task, easily prone to errors mainly because of loss of attention or fatigue of the human operator. Besides, the complexity of the task is increased by the lack of quality of typical seabed images, which are mainly characterised by blurring, non-uniform illumination, lack of contrast and instability in the vehicle motion. In this study, the development of a vision system guiding an autonomous underwater vehicle (AUV) able to detect and track automatically an underwater power cable laid on the seabed is the main concern. The vision system that is proposed tracks the cable with an average success rate above 90%. The system has been tested using sequences coming from a video tape obtained in several tracking sessions of various real cables with a ROV driven from the surface. These cables were installed several years ago, so that the images do not present highly contrasted cables over a sandy seabed; on the contrary, these cables are partially covered in algae or sand, and are surrounded by other algae and rocks, thus making the sequences highly realistic.

Key words: Features detection and tracking – Image sequences – Autonomous underwater vehicles – Pipeline inspection

1 Introduction

The feasibility of an underwater installation consisting of either cables, for power or telecommunication, or pipelines, for gas or petrol, can only be guaranteed by means of an adequate inspection programme. This programme has to provide the company with prompt information about potential hazardous situations or damages caused by the mobility of the seabed, corrosion, or human activities like marine traffic or fishing. Nowadays, those tasks of vigilance and inspection are carried out using video cameras attached to remotely operated vehicles (ROVs) controlled from the surface by trained operators. Obviously, this is a tedious task, requiring from the operator a long time of concentration in front of a console, which makes the task highly prone to errors mainly due to loss of attention and fatigue. Furthermore, undersea images possess some peculiar characteristics which increase the complexity of the operation: blurring, low contrast, non-uniform illumination and lack of stability due to the motion of the vehicle, just to cite some of them. Therefore, automating any part of the inspection process can constitute an important improvement in the maintenance of such installations, not only regarding errors, but also as far as time and monetary costs are concerned.

The special visual features that artificial objects have allow distinguishing them in natural scenarios such as the seabed, even in very noisy images. This fact makes feasible the automatic guidance of an autonomous underwater vehicle (AUV) for maintenance/inspection tasks by means of visual feedback. Following this strategy, a first approach to the problem of detecting and tracking an underwater power cable by analysing the image sequence coming from a video camera attached to an AUV was described in [11].

In this paper, an improved and optimised version is proposed. In this new approach, the initial position and orientation of the cable is computed using the first images of the sequence. Once the system has been initialised, the orientation and location computed is used to predict a region of interest (ROI) in the next image, in which the presence of the cable is highly probable in the sense of the Kalman filter. If there is strong evidence of the cable in the ROI defined in the next image, then its new orientation and position is updated and the system proceeds with the following image; otherwise, a suitable recovery mechanism is activated. In order to achieve the required real-time performance to guide the AUV, some optimisations have been incorporated into the general method of detection and tracking. Those optimisations have allowed the vision system to process images at video rate.

This study has been partially supported by the GOVERN BALEAR (BOCAIB-16,3/2/98) and project CICYT-MAR99-1062-C03-03.
Correspondence to: A. Ortiz

The system has been tested using sequences coming from a video tape obtained in several tracking sessions of various real cables with a ROV driven from the surface. These cables were installed several years ago, so that the images do not present highly contrasted cables over a sandy seabed; on the contrary, these cables are partially covered in algae or sand, and are surrounded by other algae and rocks, making thus the sequences highly realistic.

The rest of the paper consists of the following sections: Sect. 2 revises previous work on the subject; Sect. 3 describes the working scenario; a general overview of the proposed method is presented in Sect. 4, while Sects. 5, 6, 7 and 8 give the details; the configuration of the system as for the parameters involved is described in Sect. 9; results are given in Sect. 10; and, finally, Sect. 11 presents some conclusions and future work.

2 Previous work

In the literature about cable inspection, two main sensing devices can be distinguished: the magnetometer and the sonar (see [4, 7, 8], among others). The former detects the variations in the earth magnetic field induced by electrical and telecommunication cables, even when they are buried. Using this device, however, requires isolating those parts of the vehicle that generate additional magnetic fields so as not to disturb the measurements of the magnetometer. The other main sensing device, the multibeam sonar system, is mainly used in exhaustive searching missions and to detect free-spans. In those cases, the main problem consists in keeping the vehicle stabilized to improve the signal-to-noise ratio. Other systems that have also been used in cable and pipelines inspection include seismic bottom profilers [4], which combined with magnetometers allow getting a cross section of the upper bottom sediments, and PIGS [18], vehicles of small dimensions that are introduced inside the pipeline and use the present flow to move along.

In general, all these strategies need AUVs larger and more powerful than is required by the mission because of the size of the sensing devices and the consequent extra batteries [5]. By using CCD cameras, however, this problem is considerably reduced, both in cost and in AUV size. In fact, in recent years several research groups have shown the suitability of vision systems either for both navigation and mission tasks (see [10, 16, 19, 20], among others).

With regard to visual cable and pipeline tracking and inspection, several systems have been proposed so far. Matsumoto and Ito [9] developed a vision system able to follow electrical cables in underwater environments by using edge detectors, the Hough transform and some higher-level processing related to the line-like appearance of the cables. Hallset [5] presented another system able to follow pipelines also using edge detectors and the Hough transform, as well as a map of the pipeline network. At the University of Ancona a system oriented towards helping human operators in the inspection of gas and oil pipelines was also implemented [21]. In this case, the system detected the pipes and some other accessories attached to them using statistical information obtained from selected areas of the image related to the position of the cable. More recently, Balasuriya et al. proposed a system based on predicting a ROI in the image and applying the Hough transform to an edge map produced by a LoG operator [1]. An improved version using a rough 2D model of the cable appears in [2].

3 Working scenario

Just installed cables and pipelines present clearly defined features like shape and colour, which a computer vision algorithm could easily exploit to discriminate them from the surrounding environment. However, as the installation gets older, those features are modified so that more complex algorithms have to be developed. On the one hand, the mud and the sand tend to cover part of the cable, and the marine flora tends to grow on top and in the neighbourhood of it. As a result, not only does the background lack uniformity, but the apparent contours of the cable also appear broken. On the other hand, corrosion and other chemical reactions characteristic of the material give rise to random changes in colour. By way of example, Fig. 1 shows some pictures of cables installed several years ago.

Other features typical of underwater scenes also appear in Fig. 1b,d. Because ambient light decreases rapidly as a function of depth, the vehicle has to bring its own light source, i.e. spotlights. Due to the high directionality of these devices, the images usually present great differences in illumination, as shown in Fig. 1b. Additionally, the absorption and dispersion the light suffers along its propagation in the oceanic medium gives rise to blurred and low-contrast images, as shown in Fig. 1d. Finally, the fluctuations in the motion of the AUV produce changes in the orientation and position of the cable in the image, which have to be taken into account.

4 High-level description of the system

As it has been pointed out before, artificial objects present several features that distinguish them from the rest of objects present in a natural environment. In the case of the cable, given its rigidity and shape, strong alignments can be expected near its sides. This is the base strategy that the proposed system exploits to find the cable in the sequence of images.

In order to obtain the cable parameters, a segmentation step is executed. Given the contours of the resultant regions, alignments of contour pixels in a preferred orientation are determined. If among those alignments there is strong evidence of the location of the cable (two alignments with a great number of pixels lined up going from bottom to top of the image with a high degree of parallelism, even without removing the perspective effect), then the cable is considered to have been located and its parameters are computed. Otherwise, the image is discarded and the next one in the sequence is analysed.

Once the cable has been detected in the image, its location and orientation in the next one are predicted by means of a Kalman filter. In this way, the number of pixels to be processed can be reduced to a small ROI in the image. This fact lowers the probability that the system errs.

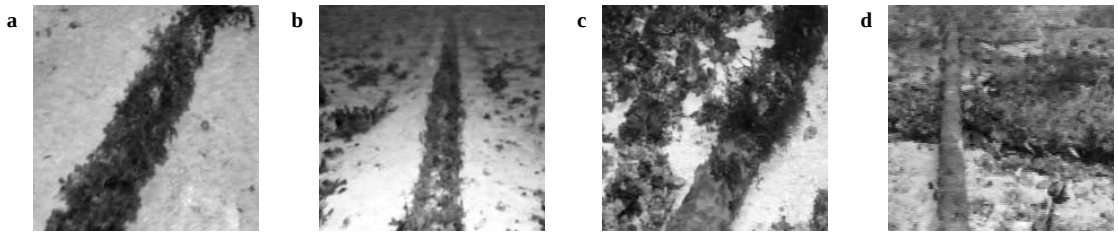

Fig. 1. Different appearances of the cable

When tracking the cable, some anomalous situations can occur, which make the system take special actions. On the one hand, if the predicted ROI is too narrow, so as not to include the whole cable, enough evidence of its location in the ROI might not be found. On the other hand, if there exists a great difference between the predicted parameters of the cable and the measured ones, then something could have gone wrong, either the prediction or the analysis of the ROI.

In any event, when any of those cases arises, it is considered as a transient failure in the analysis of the ROI, the image is discarded and no new parameters of the cable are computed. If this situation continues along too many images, then it is attributed to a failure in the prediction of the ROI. In such a case, the ROI is widened to the whole image, and the Kalman filter is reset. In other words, in front of a persistent error, the system will no longer make use of the knowledge acquired from the last processed images.

Figure 2 summarizes the way the system behaves in the different possible situations. In the flow diagram, *NonValidIMG* represents a counter of unsuccessful detections, which is reset whenever a successful detection has been achieved.

To finish this section, it must be said that the proposed system, as it is now, does not take into account the fact that the cable can be hidden under the sand, or the case when lighting is not enough due to the presence of murky water. Accordingly, if any of those situations happen for a long time, and the cable cannot be seen along the corresponding subsequence, the system can fail. This is because a minimal evidence of the presence of the cable in the image is required, and when this requirement is not met the system can consider that lack of evidence as a permanent failure and continuous resets can take place. Given the rigidity of the cable and assuming the situation does not happen for a long time, this problem can be overcome by using the prediction of the location and orientation of the cable as its location and orientation in the next frame.

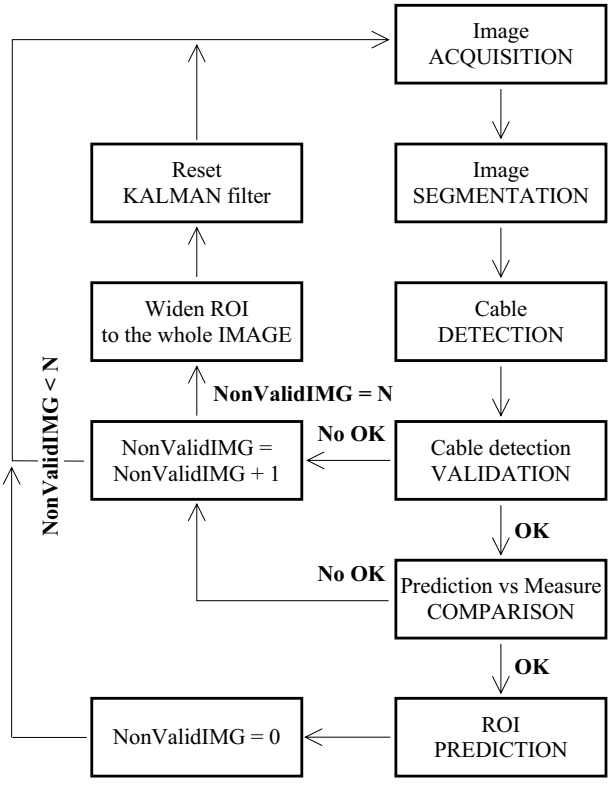

Fig. 2. Flow diagram of the system

5 Discussion about the segmentation strategy

In the oceanic medium, the appearance of artificial objects changes greatly due to several factors: bad illumination conditions, non-regular growing of flora, presence of mud and sand on top and around those objects, changes in colour due to chemical effects like corrosion, etc. This implies the segmentation algorithm should not depend, in particular, on any previous knowledge about a colour-like characteristic, but should try to look for regions in the image as dissimilar as possible amongst them in a non-supervised way. This is the main reason why the method applied in the proposed approach consists in a non-supervised clustering algorithm working over the {gray level, gradient modulus} space. A previous version of this method was suggested by Panda and Rosenfeld [14] to segment FLIR (forward-looking infra red) images, in which region contours were not clearly defined.

Transforming the image into the {gray level, gradient modulus} space consists in building a bidimensional histogram where one horizontal axis corresponds to gray level, the other horizontal axis corresponds to a digital approximation of the modulus of gray-level gradient, and for every combination (gray level, gradient modulus) the vertical axis is the number of pixels in the image having that gray level and that gradient modulus.

In the case of several objects with different gray levels, the ideal histogram should look like Fig. 3a. In effect, if the camera is located rather far from the scene, the interior of any object in the image has gradient near zero, so that pixels in the interior zones would be located in the lower part of the histogram, with regard to gradient. If, for instance,

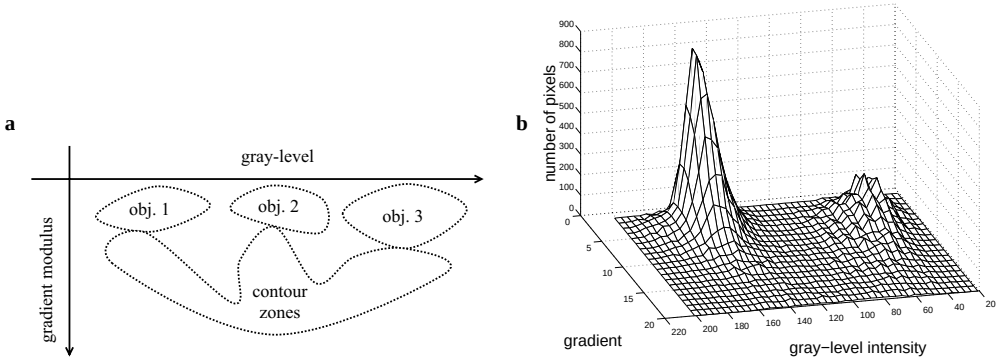

Fig. 3. a Ideal bidimensional histogram; **b** bidimensional histogram of a real image (Fig. 1a)

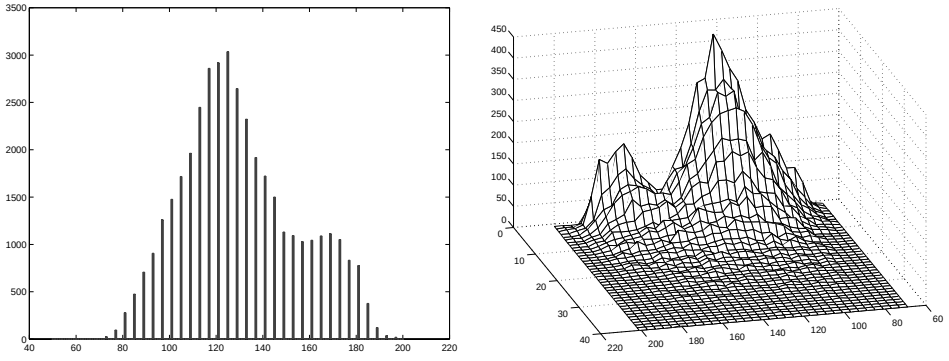

Fig. 4. Comparison between the unidimensional and the bidimensional histogram of a real image (Fig. 1d)

the scene consisted of two objects and the background, with different identifying gray levels, the lower zone of the histogram would present, thus, three clusters corresponding to the interiors of those objects in the image and the background. Border pixels between such objects, however, would be located in zones of higher gradient joining the previous clusters in the fingers-like fashion shown in Fig. 3a. By way of example, Fig. 3b presents the histogram corresponding to the real image appearing in Fig. 1a.

In clear images like Fig. 1a, the histogram shows clearly separated clusters. Obviously, the more separated those groupings are, the better are the resultant regions of the segmentation step. The fact that the clusters appear more or less separated depends not only on the image and the level of noise, but also on the gradient operator. For instance, the histogram shown in Fig. 3b was obtained using the Sobel operator.

Figure 4 compares a gray level histogram and the corresponding bidimensional histogram for image in Fig. 1d. As it can be seen in the figure, the gray level histogram is mostly unimodal, so that traditional thresholding algorithms [13] would not even be able to distinguish two classes. At the bidimensional histogram, however, at least two clusters can be clearly distinguished.

Once the histogram has been built, a grouping process is applied over its non-null cells in order to determine the aforementioned clusters. Among the different grouping algorithms that can be found in the literature, two of them have been tested: the minimal-spanning tree algorithm, as a hierarchical method, and the K-means algorithm, as a partitional method [6]. In the former, the two nearest classes are joined at every iteration, and the process stops when the desired number of final classes is reached; in the end, a hierarchy of levels or dendogram is obtained. In the latter, an initial partition is established by defining K class-centroids over the working space and the existing classes are joined to the nearest centre; the centroids are computed again and a new iteration is executed, until convergence is reached. As for the distance function selected, experimental results on several of them (see [3] for a thorough discussion) have shown that the best results are obtained using the Euclidean distance.

To finish this section, Fig. 5 shows segmentations generated by both clustering methods for all the images in Fig. 1. In both cases, the number of final classes, k_f, has been fixed to three. Furthermore, in the case of the K-means algorithm, the required first partition is obtained by defining as initial centroids the smallest, the median and the largest gray level, with gradient zero in all cases. This election stems from the fact that no information about the gray level of the expected objects can be used due to the discussed peculiarities of underwater images.

With regard to Fig. 5, it is important to note that, although in some segmentations there is not a single region wholly matching the cable, the contours derived from both methods give a clear idea where it is in the image. The hierarchical method, however, seems to produce better segmentations, at a similar total frame processing time. Probably this is due to the quasi random selection of initial centroids in the K-means segmentation version. Since this behaviour has also been observed when processing image sequences, better results are expected from the hierarchical method. This is the reason why the latter has been finally selected.

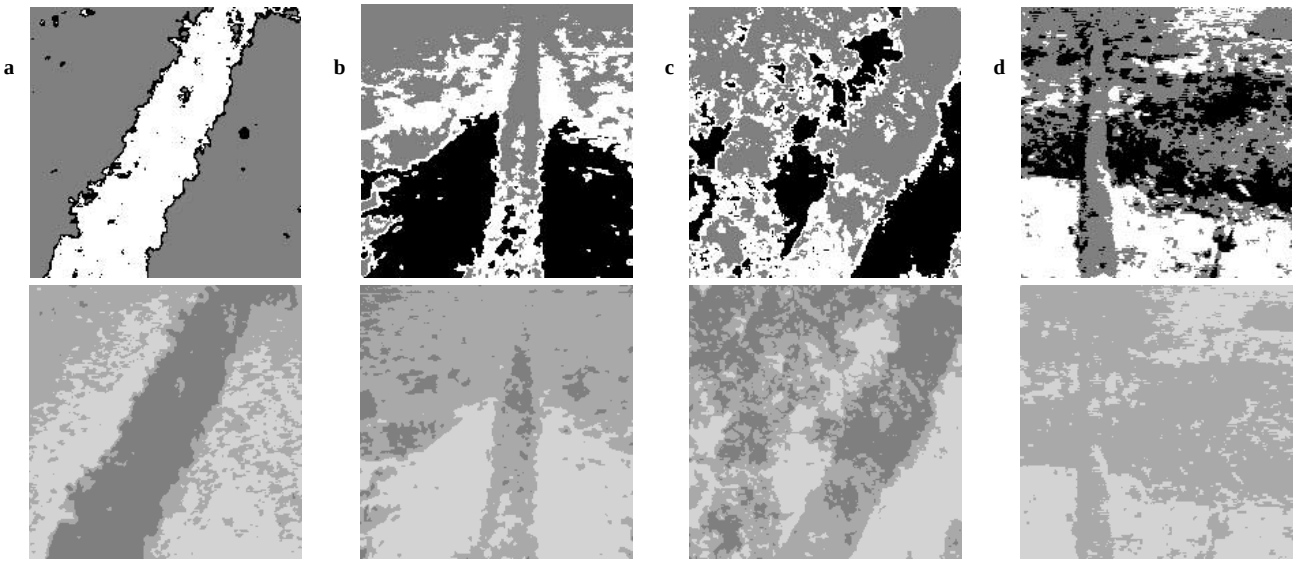

Fig. 5. Segmentation in three classes for all the images in Fig. 1 (*top row* shows minimal-spanning tree results, and *lower row* shows K-means results)

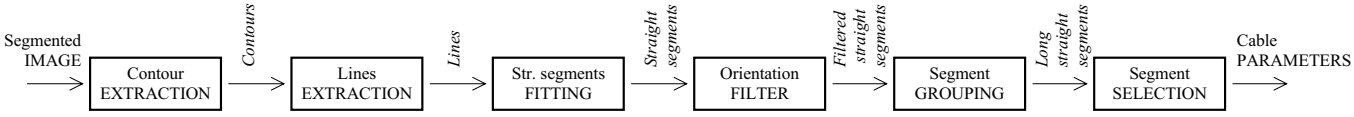

Fig. 6. Flow diagram of the cable detection step

6 Computation of the cable parameters

Once the image has been segmented, the system proceeds to locate the cable, executing the tasks enumerated in Fig. 6. This step is carried out from the contours of the segmented image, by looking for lines that can belong or be near the sides of the cable. In this context, a line is defined as a set of connected contour pixels not including branches. Moreover, as it is perfectly reasonable to expect the cable to cross the image from bottom to top – this is just a matter of navigation control – some tasks of the detection step will use this assumption for filtering purposes to reduce the probability of erroneous detection. Note that this assumption implies the cable should appear vertical or near-vertical in the image.

Lines are obtained by scanning the segmented image from bottom to top. The direction of scanning is important as the lower part of the image tends to be clearer than the upper part when the camera is not oriented towards the seabed, due to the properties of light propagation undersea (e.g. Fig. 1b). Once a contour pixel (i, j) has been found, adjacent pixels are selected according to the preference matrix shown in equation (1).

$$\begin{bmatrix} 3 & 1 & 2 \\ 5 & (i,j) & 4 \\ 0 & 0 & 0 \end{bmatrix} \tag{1}$$

The numbers indicate preference, where the lower the number, the higher the preference, except for zero, which represents a forbidden selection. Therefore, lines in vertical directions are favoured against horizontal or curving lines, according to the above-mentioned assumption. When, for a given contour pixel, there is no adjacent pixel in the preferred orientations, the process of tracking the line finishes and a new one starts by resuming the scanning of the image from the point it was left.

A straight segment fitting task follows next. This process can be seen as a low-pass filter to remove noise due to both the redefinition of the cable contours caused by the proliferation of flora on top of and by the cable, and due to the processes of image acquisition and segmentation. An eigenvector-based method has been used in the fitting [3]. As the fitting error can become large in some cases, a control procedure is executed after each fitting. It is as follows:

- For each point p_i belonging to a line L, its orthogonal distance to the fitted straight segment S, $d(p_i, S) \geq 0$ is computed.
- If $d(p_j, S) = \max\{d(p_i, S) | p_i \in L\} \geq k_e$, then L is split into two at the point of greatest local maximum error that is not an end of the line. In this way, the pathological case of always splitting the line at its ends is avoided, as it is shown in Fig. 7.

The resultant set of straight segments is filtered according to their orientation in the image. As it is supposed this orientation is vertical or near-vertical, all the segments whose orientation do not belong to a suitable angular interval $k_o \pm \Delta = 90^\circ \pm \Delta$ are rejected. Once the system has been initialised, the interval $k_o \pm \Delta$ is adapted to the previous orientation found for the cable.

Subsequently, a co-linearity analysis is applied to the set of straight segments obtained in order to join the segments that can be considered as originally belonging to the same straight contour. As an example of the analysis performed, consider the set of segments that have passed the orientation filtering process (see Fig. 8). For each straight line segment S_i under analysis, a new straight line LS_i is calculated using

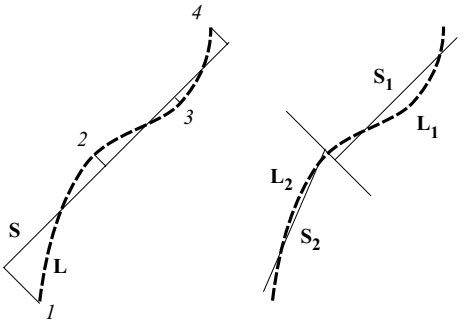

Fig. 7. Splitting of a line L during the straight segment fitting process. *1*, *2*, *3* and *4* are all local maxima of $d(p_i, S)$, and *1* and *4* are the ends of the line too. L is split by the greatest local maximum that is not an end, *2*, and the process is repeated for L_1 and L_2

again the eigenvector straight line fitting method. This time, the points used in the fitting, $\{p_j\}_i$, are those contour points corresponding to the straight segments that completely fall within a strip-shaped region whose width is ω and that is aligned with S_i. ω is the tolerated co-linearity error.

The last task of the detection step consists in choosing a pair of long, straight segments that are likely to correspond to the sides of the cable. The main requirements for such pairs are a certain minimum length for each segment, a high degree of parallelism and a certain separation between each segment. Therefore, for every possible pair of long, straight segments, the above requirements are checked and, in the cases where they are met, a score equal to the sum of their lengths is assigned to the pair. The pair with the highest score is selected and used to obtain the parameters of the cable. If there does not exist any pair satisfying the requirements, it is considered that there is not enough evidence for the presence of the cable in the image.

By way of example of the whole process of detection of the cable, Figs. 9 and 10 show the respective intermediate and final results for images Fig. 1b,d. The values of the main parameters have been fixed to: straight line fitting error k_e = 2, orientation accepted $k_o \pm \Delta = 90° \pm 30$, co-linearity error ω = 5, minimum length for each pair of long, straight segments to be considered for selection equal to the average length of all the long, straight segments. More final results for images of higher complexity can be found in Fig. 11.

7 Tracking strategy

The cable-tracking strategy is based on the hypothesis that the cable parameters are not going to change too much from one image to the next. Therefore, once the cable has been detected in the image sequence, the computed position and orientation are used to predict the new parameters in the next image. This prediction allows introducing a further checking point in the sense of comparing predicted parameters with computed parameters, reducing, at the same time, the image area where to look for the cable and, thus, increasing the probability of success. This area is the above-mentioned region of interest (ROI). As it has been said before, in case the system is not able to find enough evidence of the cable in the ROI, the recovery mechanism previously described is activated.

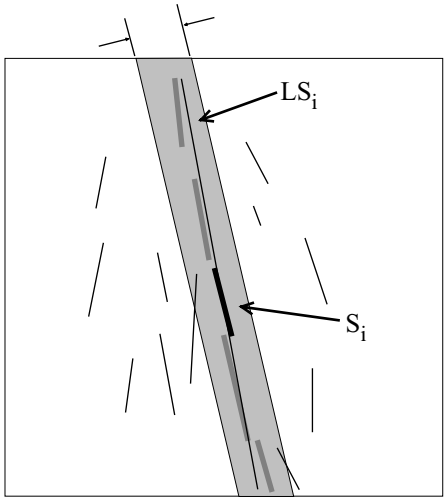

Fig. 8. Co-linearity analysis

To predict the cable parameters, the system makes use of a linear Kalman filter for every side of the cable. The reason why the linear version of the filter was chosen is because there were no data available about the dynamics or the motion of the vehicle carrying the camera, so that, as the simplest option, the linear variant was tested, giving fairly acceptable results. However, as soon as a vehicle is available for the mission of tracking the cable, the filter will be adapted to it to improve its performance. The state vector X for each side contains its position x and its orientation α with regard to the horizontal axis of the image. Equation (2) expresses the filter models for the left and right sides. In both sets of equations, v represents the process noise and w corresponds to the measurement noise.

$$\begin{aligned} X_L &= (\alpha_L, x_L) \\ X_L(t+1) &= X_L(t) + v_L(t) \\ Z_L(t+1) &= X_L(t) + w_L(t) \end{aligned}$$
$$\begin{aligned} X_R &= (\alpha_R, x_R) \\ X_R(t+1) &= X_R(t) + v_R(t) \\ Z_R(t+1) &= X_R(t) + w_R(t) \end{aligned} \qquad (2)$$

To compute the process noises v_R and v_L, several real sequences were manually analysed and the differences between consecutive frames in the cable positions and orientations were computed. At the end of this procedure, an estimation of the covariance matrix of both noises was available. As for the measurement noises w_R and w_L, the system was faced against noisy synthetic sequences and the deviations between the real orientations and positions of the cable and the measured ones were determined. The corresponding covariance matrixes were finally obtained from those deviations.

Finally, the ROI for the next image is computed as it is indicated in Eq. (3), where $\hat{r}_L$ and $\hat{r}_R$ represent the predicted straight sides of the cable, and k_L and k_R are tolerance factors included in the ROI. Best results have been obtained by

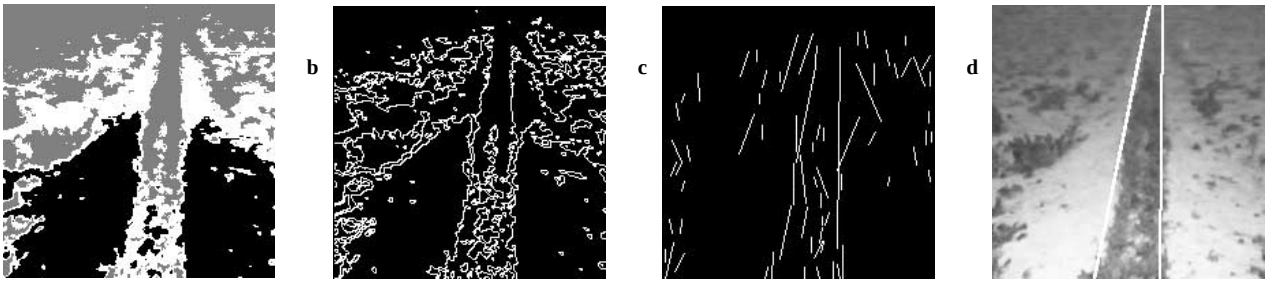

Fig. 9. Intermediate and final results for image Fig. 1b: **a** segmented image; **b** contour image; **c** straight segments after the orientation filter; **d** detected sides of the cable

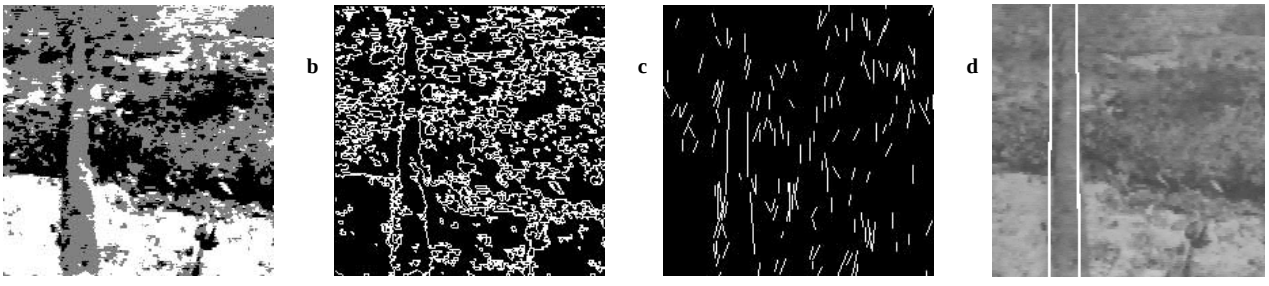

Fig. 10. Intermediate and final results for image Fig. 1d: (a) segmented image; (b) contour image; **c** straight segments after the orientation filter; (d) detected sides of the cable

computing them as twice the difference between the computed and the predicted position of each side.

$$ROI(t+1) = [\hat{r}_L(t+1) - k_L(t+1), \hat{r}_R(t+1) + k_R(t+1)] \quad (3)$$

8 Optimisations of the proposed method

The proposed method of detection and tracking the cable in a sequence of images is highly time consuming. In order to achieve a real-time performance, some optimizations have been applied to the algorithm. They are detailed as follows:

- First, among the several gradient operators proposed in the computer vision literature – e.g. forward differences, centred finite differences, Roberts, Sobel, Prewitt, and Canny (see [15, 17] for a discussion on the subject) – the one which has given the best results, both as a digital approximation to the real gradient of a gray-level image and regarding computation time, has been the Sobel operator. As it is an operator involving weights that are power of 2, it can be programmed in a very efficient way using integer computations, which meaningfully accelerates the execution.
- Second, as the clustering stage is, in the worst case, an $O(n^3)$ process, where n is the number of initial classes to group, diminishing n as much as possible dramatically contributes to reducing the computation time. After some experimental work, two optimisation strategies have been introduced: (1) reduce the resolution of the histogram by a factor k_h; and (2) reduce the classes to group to a subset consisting of the k_c most meaningful ones, the ones representing the highest number of pixels. The best results have been obtained for k_h = 8 and selecting the k_c = 20 classes representing the largest number of pixels. On the one hand, this means that the limits of the gray-level axis of the histogram are 0 and $\frac{256}{8} - 1 = 31$, as well as the gradient axis; on the other hand, once the k_c most meaningful classes have been grouped into the k_f = 3 final classes, the remaining classes have to be added a posteriori to the nearest final class to complete the segmentation. The experiments performed have shown that both optimisations imply a great reduction in the computation time without degrading significantly the results of the segmentation.
- Third, during the detection step, the execution time of every task is a process $O(n)$ or $O(n^3)$, where n is the number of lines or the number of straight segments. As most of them are unlikely to correspond to the cable contours, and the ones belonging to the latter are expected to have higher lengths, a filtering process is applied to the sets of lines and straight segments so that only the k_s longest ones survive to the next task. Best results have been obtained for k_s = 100.
- Finally, the prediction of a ROI for every image in the sequence gives rise to an important reduction in the execution time, as it means processing up to 20% of the original number of pixels of any image, including the tolerance area added by means of k_L and k_R in Eq. (3).

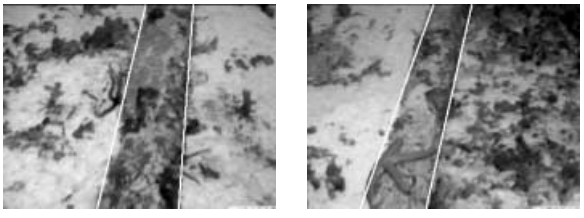

Fig. 11. Final results for two images of high complexity. Both pictures show the detected sides of the cable

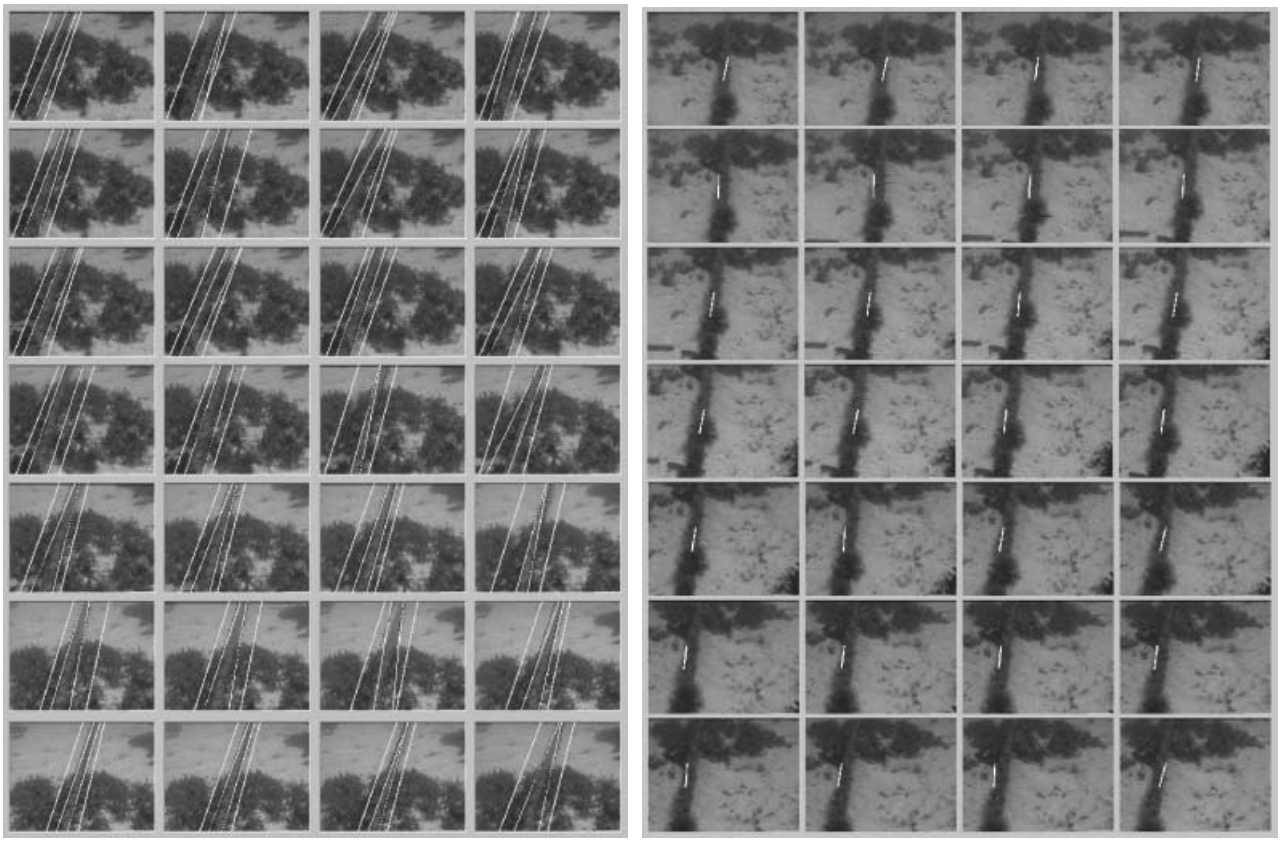

Fig. 12. Results for an excerpt of sequence 1. In the *left image*, the ROI appears superimposed together with the reconstructed sides of the cable (the *black lines* correspond to the predicted sides). In the *right image*, the *white line* represents the axis of the cable and thus a possible command to the AUV controller

9 System configuration

As it has been seen throughout the paper, a relatively high number of parameters determine the behaviour of the system when processing either a single image or a sequence. However, it has not been excessively difficult to select a stable set of values for which the system behaves correctly across the different scenarios. A more detailed discussion about how to set them up can be found in the following, although some examples of values of some of these parameters have already been given to illustrate the results of related tasks. By way of summary, Table 1 puts them together.

The values chosen come from experimental observations of the behaviour of the system against different test images and sequences considered as representing the whole range of complexity. By way of example, at the low-complexity end we can find images like in Fig. 1a,b (i.e. cables over a sandy seabed), while those in Fig. 1c,d and, even more, the images in Fig. 11 belong to the high-complexity end (i.e. cables over rocks and algae, and/or a high degree of cable occlusion).

Parameters k_h, k_c and k_s are related with the optimisations introduced in the system to attain real-time performance, and they have already been discussed in Sect. 8. Therefore, the values selected correspond to the ones for which a higher reduction in the execution time were achieved without degrading the results of the corresponding stage (i.e. the quality of the segmentation in the case of k_h and k_c, and the quality of the detection in the case of k_s).

k_L and k_R are also related with the reduction in the time of execution: the lower k_L and k_R are, the better. However, very low values for k_L and k_R also reduce the probability of locating the cable within the ROI. As has been stated in Sect. 7, a value of twice the difference between the predicted side of the cable ($\hat{x}_L$ and $\hat{x}_R$) and the detected one (x_L and x_R) is enough.

As for k_f, in some images it is better to stop the clustering when two classes are reached (say for a cable over a sandy seabed), but, in more complex images, a classification in two classes gives rise to very few contours. Consequently, poor detections of the cable are given. However, k_f = 3 is adequate whatever the image complexity.

Finally, k_e and ω depend on the amount of redefinition of the sides of the cable due to the proliferation of seaweed and algae on top of it and near its base. That is to say, the older the installation, the higher these values must be.

10 Tracking results

To test the system, both synthetic and real sequences have been used. The experiments performed with the former have shown the robustness of the proposed detection and tracking method against noise, while experimental work with the latter have permitted testing the system with highly realistic and complex situations. However, due to lack of space, only

Fig. 13. Results for excerpts of sequences 2 (*upper row left*), 3 (*upper row right*), 4 (*lower row left*) and 5 (*lower row right*)

Table 1. System parameters

Stage	Symbol	Meaning	Typical value
Segmentation	k_h	Histogram resolution reduction	8
	k_c	Most important classes to be grouped	20
	k_f	Number of final classes	3
Detection	k_e	Fitting error	2 image units
	ω	Co-linearity error	5 image units
	k_s	Limit in the number of segments and straight lines	100
Tracking	k_L	ROI left-side tolerance factor	$2 \times \lvert\hat{x}_L - x_L\rvert$ image units
	k_R	ROI right-side tolerance factor	$2 \times \lvert\hat{x}_R - x_R\rvert$ image units

Table 2. Image sequence results

Sequence	Length (frames)	Frame rate achieved	Wrong detections	Success rate
1	248	35.71 f/s or 28 ms/f	21	92%
2	499	30.30 f/s or 33 ms/f	20	96%
3	409	25.64 f/s or 39 ms/f	69	83%
4	172	28.57 f/s or 35 ms/f	2	99%
5	129	28.57 f/s or 35 ms/f	3	98%
Average	1457	29.75 f/s or 33.6 ms/f	115	92%

results for real sequences will be shown in the following. For a detailed description of the experiments with synthetic sequences, the reader is referred to [12].

As for the real sequences, they come from several ROV sessions recorded on video tape. During those sessions, a ROV was manually guided by an operator on the surface. This fact means the output of the vision system has not been used to correct the vehicle's course, whereby there are sudden changes in direction and altitude due to the commands sent by the operator. Certainly, the movement would be softer and of less magnitude if the vehicle was controlled in an automatic way, which would also ease the task of locating the cable in the sequence.

Five sequences of different complexity have been selected from the mentioned video tape. Although they are not very lengthy, due to the impossibility of finding long sequences in the recorded material, they cover the whole range of complexity – steep gradients in illumination, low contrast and blurring, objects overlapping the cable, instability in the vehicle motion, etc. – as will be seen below. Table 2 lists important information about every sequence. The success rate in the table refers to those images for which the ROI wholly includes the cable and the system has been able to determine correctly its location. As for the execution times, they do not include the acquisition time of the frame grabber, as this time depends on the particular device used. Finally, the column named *frame rate achieved* gives an idea of the mean execution time for every sequence, or, conversely, the number of frames per second the system could sustain. All the tests have been run on an 350 MHz AMD K6-2 machine executing Windows NT v4.0, and the resolution of the images was half-NTSC (320×240 pixels).

Figure 12 left shows results for an excerpt of sequence 1 with the ROI superimposed. The excerpt presented is interesting in the sense that there is little evidence of the cable in the image due to the large algae zone that surrounds it. In fact, at the beginning of the excerpt, the system has some problems to locate the cable inside the ROI, although it recovers quite well after two frames. Figure 12 right shows results for another excerpt of sequence 1, in which the white central line represents the computed axis of the cable, and its parameters would be, after a proper coordinates transformation, the sort of command the navigation controller of the AUV would receive. As it can be observed, the system always returns the cable axis within the cable region of the image, so it can be said the position of the cable is correctly detected every time. The orientation, however, is mostly affected by the noise present in the image in the form of algae surrounding the cable and thus breaking its original contours. This excerpt also shows how the system is able to tolerate some sudden movements of the vehicle: during the first images, the pitch angle of the ROV allows the camera to capture the algae zone that appears in the upper part of those images; as the ROV moves forward, the operator orders it to increase the pitch angle towards the seabed, which makes the algae zone disappear; in the last images, however, the same algae zone appears again as the pitch angle has been reduced. During the whole process, however, the cable has not been lost.

Likewise, Fig. 13 top left presents the same sort of results as Fig. 12 right but for an excerpt of sequence 2. It is a sequence similar to sequence 1, but with a high degree of gradient in the illumination, together with the fact that the cable appears thicker in the image as the camera is closer to the scene. Results for image sequences 2 and 3 can be seen, respectively, in Figs. 13 top right and bottom left. Both sequences are far more complicated than sequences 1 and 2 due to the high degree of overlapping over the cable and the lack of contrast between cable and seabed. Furthermore, Fig. 13 bottom left shows how the system behaves with a nearly hidden cable. Finally, Fig. 13 bottom right corresponds to sequence 5. As it can be seen, the height of the vehicle along the sequence changes, as in sequence 1, and a high gradient in the illumination is again present.

11 Conclusions and future work

A vision system for real-time, underwater cable tracking has been presented. Using only visual information, the system is able to locate and follow a cable in an image sequence, overcoming the typical difficulties of underwater scenes: blurring, light attenuation, inhomogeneous lighting, flora and fauna overlapping the object of interest, etc. Several sequences coming from a video tape obtained in several tracking sessions of various real cables with a ROV driven from the surface have been used to test the system. These cables were installed several years ago, so that the images do not present highly contrasted cables over a sandy seabed; on the contrary, these cables are partially covered in algae or sand, and are surrounded by other algae and rocks, thus making the sequences highly realistic. The mean success rate that has been achieved is above 90% for a frame rate of more than 25 frames/s.

This study is included in a more ambitious project in which the inspection of underwater installations in connection with the presented tracking system is the main concern. While in the tracking step, the defects and strange situations should be detected and stored for an off-line analysis by an expert. Therefore, a means for locating the defect once analysed should be provided, e.g. by fusion of GPS, dead-reckoning and visual data. A more accurate positioning method would link the AUV via acoustic sensors to a transmission buoy connected to GPS sources.

Acknowledgements. The authors of this study wish to thank GESA for providing them with the cable image sequences used to test the system, and the anonymous reviewers for their insightful comments and helpful suggestions on the earlier versions of this paper.

References

1. Balasuriya B, Takai M, Lam W, Ura T, Kuroda Y (1997) Vision based autonomous underwater vehicle navigation: underwater cable tracking. In: Oceans '97 MTS/IEEE Conference Proceeding, vol 2, pp 1418–1424
2. Balasuriya B, Ura T (1999) Multi-sensor fusion for autonomous underwater cable tracking. In: Oceans '99 MTS/IEEE Riding the Crest of the 21st Century, vol 1, pp 209–215
3. Duda RO, Hart PE (1973) Pattern Classification and Scene Analysis. Wiley, New York
4. Egeskov P, Bech M, Bowley R, Aage C (1995) Pipeline inspection using an autonomous underwater vehicle. In: Proceedings of the 14th International Conference on Offshore Mechanics and Arctic Engineering, vol 5, pp 539–546
5. Hallset JO (1996) Testing the robustness of an underwater vision system. In: Laplante P, Stoyenko A (eds) Real-time imaging: theory, techniques, and applications. IEEE Press, pp 225–260
6. Hartigan J (1975) Clustering algorithms. Wiley, New York
7. Ito Y, Kato N, Kojima J, Takagi S, Asakawa K, Shirasaki Y (1994) Cable tracking for autonomous underwater vehicle. In: Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology, pp 218–224
8. Iwanowski MD (1994) Surveillance unmanned underwater vehicle. In: Proceedings of IEEE Oceans, pp I116–I119
9. Matsumoto S, Ito Y (1995) Real-time vision-based tracking of submarine cables for AUV/ROV. In: Proceedings of MTS/IEEE Oceans, pp 1997–2002 Part 3 of 3
10. Nguyen HG, Kaomea PK, Heckman PJ Jr (1988) Machine visual guidance for an autonomous undersea submersible. In: Proceedings SPIE Underwater Imaging, vol 980, pp 82–89
11. Ortiz A, Oliver G, Frau J (1997) A vision system for underwater real-time control tasks. In: Oceans '97 MTS/IEEE Conference Proceeding, vol 2, pp 1425–1430
12. Ortiz A, Simó M, Oliver G (2000) Real-time image sequence analysis for tracking an underwater power cable. Technical Report 2/2000, Departament de Matemàtiques i Informàtica (Universitat de les Illes Balears)
13. Pal NR, Pal SK (1993) A review on image segmentation techniques. Pattern Recogn 26(9):1277–1294
14. Panda DP, Rosenfeld A (1978) Image segmentation by pixel classification in (gray level,edge value) espace. IEEE Trans Comput C-27(9):875–879
15. Parker J (1997) Algorithms for image processing and computer vision. Wiley, New York
16. Santos-Victor J, Sentieiro J (1994) The role of vision for underwater vehicles. In: Proceedings of the IEEE Symposium on AUV Technology, pp 28–35
17. Schalkoff RJ (1989) Digital image processing and computer vision. Wiley, New York
18. Wearden T (1992) Transducers and sensors for pipeline inspection vehicles. Control Syst J 9(8):
19. Xu X, Negahdaripour S (1997) Vision-based motion sensing for underwater navigation and mosaicing of ocean floor images. In: Oceans '97 MTS/IEEE Conference Proceeding, vol 2, pp 1412–1417
20. Xu X, and Negahdaripour S (1999) Automatic optical station keeping and navigation of a ROV; sea trial experiments. In: Oceans '99 MTS/IEEE Riding the Crest of the 21st Century, vol 1, pp 71–76
21. Zingaretti P, Tascini G, Puliti P, Zanoli S (1996) Imaging approach to real-time tracking of submarine pipeline. In: Proceedings SPIE Electronic Imaging, vol 2661, pp 129–137

Alberto Ortiz received his degree in Computer Science from the University of the Balearic Islands (UIB) in 1992. He joined the Mathematics and Computer Science Department of the UIB in 1992 and has been a faculty member since then. He is author and co-author of several international publications related with computer vision and robotics and has participated in several projects related with VLSI circuits design, fault-tolerant systems, and computer vision and robotics. Now, he is a member of the Systems, Robotics and Vision (SRV) group of the UIB. His present research interests include land and underwater computer vision, land and underwater robotics and real-time systems.

Miquel Simó received his degree in Computer Engineering from the University of the Balearic Islands in 2000. He has participated in projects related with underwater computer vision and robotics. He is now a Software Engineer.

Gabriel Oliver received a degree in Physics from the Autonomous University of Barcelona in 1985. He earned his Ph.D. in Computer Science from the Polytechnic University of Catalonia in 1993. His major research interests are real-time vision and control architectures for autonomous mobile robots. He has been the leading researcher of projects granted by the local administration and the Spanish Scientific Council (CICYT), both sorts being related to underwater robotics. He is currently a senior lecturer at the Department of Mathematics and Computer Science at the University of the Balearic Islands where he is the Director of the Systems, Robotics and Vision (SRV) group of the University of the Balearic Islands (UIB).