Augmented Scene Modeling and Visualization by Optical and Acoustic Sensor Integration

Andrea Fusiello and Vittorio Murino*

February 13, 2003

Abstract

In this paper, underwater scene modeling from multisensor data is addressed. Acoustic and optical devices aboard an underwater vehicle are used to sense the environment in order to produce an output that is readily understandable even by an inexperienced operator. The main idea is to integrate multiple-sensor data by geometrically registering such data to a model. The geometrical structure of this model is a-priori known but not *ad hoc* designed for this purpose. As a result, the vehicle pose is derived, and model objects can be superimposed upon actual images, thus generating an augmented-reality representation. Results on a real underwater scene are reported, showing the effectiveness of the proposed approach.

Keywords Augmented reality, enhanced vision, multisensor integration, acoustic imaging, teleoperation, model-view registration, underwater applications.

*Andrea Fusiello and Vittorio Murino are with the Dipartimento di Informatica, University of Verona, Italy. E-mail:{fusiello,murino}@sci.univr.it

1 Introduction

Badly structured environments, like the underwater world, are difficult to perceive and understand. Nevertheless, underwater scene exploration is an expanding research field, linked to the great interest in monitoring the evolution of the subsea flora and fauna and to the sustainable exploitation of such an environment. To this end, the use of multiple sensors is typically necessary, but the related data integration is critical.

This paper describes the design and implementation of an augmented-reality system to support the human operator of an underwater Remotely Operated Vehicle (ROV). The three-dimensional (3-D) synthetic models of objects of interest are overlaid onto a real image to generate an augmented-reality representation, thus improving the perception and understanding of the environment (so as to facilitate the vehicle navigation) and the effectiveness of the exploration.

Two sensing channels (optical and acoustic) are mostly used underwater. Typically, optical images are easier to interpret by the human operator, but the underwater visibility range is very limited due to low illumination and the presence of clutter (even though special sensing configurations under investigation, like range gated imaging systems and laser-based devices, can overcome such limitations). On the other hand, 3-D acoustic data are not affected by illumination problems but are more difficult to understand for the human operator. From these considerations, it appears sensible to try to integrate the two channels in order to exploit the best of both so as to compensate for their drawbacks.

Augmented Reality (AR) supplements reality by allowing the user to perceive

the real world, with virtual objects superimposed upon it or merged with it. Virtual objects convey information that the user cannot directly detect with her/his own senses. A comprehensive review of AR can be found in [1].

In order to make synthetic graphics appear in the proper place (for example, like a wire-frame outline superimposed on top of the corresponding real-world object), it is necessary to know exactly the pose (i.e., position and orientation) of the camera in the real world. This is the so-called *registration problem*, which is still a challenge if one cannot rely on a tracking system mounted on the camera.

Video-based approaches, where a real image and the graphic overlay are combined by using a video camera and a computer (as opposed to optical approaches, where the overlay is obtained by means of a see-through display), can use computer vision techniques to aid registration [2]. Since video-based AR systems have a digitized image of the real environment, it is possible to enforce registration of the model onto the view of the real world. This constitutes a "closed-loop" approach, as the digitized image provides a mechanism for bringing feedback into the system. In video-based systems, the same video camera as used to capture the video image serves as a tracking device as well. Moreover, the pose calculation is accurate on the image plane, thereby minimizing the error on the perceived image alignment.

Considered from another viewpoint, registration can also be accomplished via scene modeling, which is a problem extensively investigated nowadays, especially for robotic purposes. In [3], [4], and [5], a laser range-finder is used to acquire a depth map of a real object. The system is able to align the real and virtual depth maps, thus providing the information needed for registration. In [6], range

data are segmented into planar and quadric surfaces. Then, the operator selects object models contained in a region of interest and a matching phase is carried out to recognize object models and estimate their poses in the scene. This work is improved in [7], in which model objects and actual objects are generally represented by surface meshes; as a result, the recognition and pose estimation phases are more reliably performed by using *spin images*. Accurate segmentation of range images for industrial pipe modeling is carried out in [8]. Segmentation is performed by fitting data to geometric models chosen from among a small set of primitives: plane, cylinder, torus, cone, and sphere. The principal curvatures are estimated in a robust way together with their centers, and heuristics are used to ensure pipe continuity (e.g., setting a connectivity threshold).

All the cited works consider laser range-finder data that are affected by noise only to a limited extent, as compared with 3D data acquired by an acoustic sensor. In the latter case, speckle noise degrades the image much more and injects a high percentage of outliers. For these reasons, our approach must be more robust than those used in the above-described applications.

Instead of using them as alternative sensors to video cameras, laser range-finders can be used in addition to video cameras for better reconstruction and to support the registration process. Indeed, fusion of range and optical data is recognized to be important in many respects, and much research is oriented in this direction.

In [9], registered range and optical aerial images are used to detect and reconstruct buildings. In [10], a Markov Random Field (MRF) model is proposed for the

fusion of registered range and intensity images for the purpose of image segmentation. A similar method for the fusion of range and intensity images is adopted in [11], where edge detection, semantic labeling, and surface reconstruction are integrated into a single framework.

Although fusion and integration of different kinds of data are a matter of active research [12, 13], to the best of our knowledge our approach to sensor integration and data fusion contains some original ideas, and no similar works are available in the literature. A method was recently proposed [14] that resembles our approach, although with substantial differences. In [14], a pipeline methodology for editing a real scene acquired by an optical camera and a laser range-finder is presented. It assumes a higher degree of control of the scene by requiring manual intervention and the use of artificial fiducials spread over the whole scene. The final goal of recovering the 3-D scene modeling is achieved 1) by segmenting the 3-D image with an interactive graph-based technique, and 2) by deriving photometric information via the estimation of the optical camera pose with the aid of calibration targets.

Our approach proposes to automatically fuse optical and range data to recognize and estimate the poses of 3-D objects for the purpose of obtaining an augmented-reality representation. We aim to locate model objects present in a cluttered scene and to facilitate human interpretation by displaying such objects on the real images in the correct positions and orientations. Acoustic and optical underwater data are first processed separately in order to estimate the positions of the objects present in the scene. In this way, the relative poses of acoustic and optical cameras are estimated on line and actual data integration is achieved.

This implicit calibration process is carried out without using *ad hoc* designed objects or a particular sensorial set-up, and the result of this phase is that 3-D information is registered to the optical image pixels, with a great advantage in terms of precision and scene comprehension.

It is worth noting that the operative conditions of the underwater environment are very badly structured, and no accurate control or positioning can actually be performed by the sensor devices aboard a vehicle.

The proposed system is composed of a set of modules to process acoustic and optical data; the modules already contain *per se* some novel aspects and solutions necessary to deal with the very uncertain, noisy nature and different resolutions of the two types of data. However, the most original contribution consists in the development of a system able to integrate different sensors and fuse different kinds of data in *numerical* form, dealing with very sparse and noisy range data.

The rest of the paper is organized as follows. After an overview of the global system in Section 2, the acoustic sensing phase and the related data processing are addressed in Section 3. In Section 4, the description of the optical sensory and processing channel is provided. The integration phase is illustrated in Section 5, and two examples showing the method's performance on real data are given in Section 6. Finally, conclusions are drawn in Section 7.

2 System Overview

The application scenario consists in an ROV approaching an oil rig whose geometrical model is given in a descriptive language (e.g., Virtual Reality Modeling

Language, VRML). The ROV is equipped with an optical and an acoustical camera located at fixed but unknown relative positions. The optical camera provides 2-D intensity images, whereas the acoustical one provides an image consisting of a set of 3-D points [15]. These images are not registered and are only partially overlapped, as the points of view and the view frusta are different for the two sensors.

The oil rig is a complex structure of connected pipes, and the goal of the system is to identify and locate the joints, thereby obtaining the position of the ROV in a world reference frame.

The system is subdivided into two data-processing threads that are related to the two sensory channels, and that eventually merge in the integration stage (see Fig. 1). Each thread is composed of several modules devoted to object recognition and pose estimation. The acoustic data-processing thread includes the following modules:

- filtering, to reduce noise and eliminate spurious points;
- segmentation, to determine the most significant regions;
- classification and reconstruction, to label the regions according to their shapes and to carry out a rough reconstruction on the basis of the geometrical features estimated by the classification module;
- recognition and model-view registration: a matching phase is first performed between classified regions and a model base to identify the observed object models; the result of this phase is then used to refine the reconstruction by an accurate registration process and to estimate the relative pose.

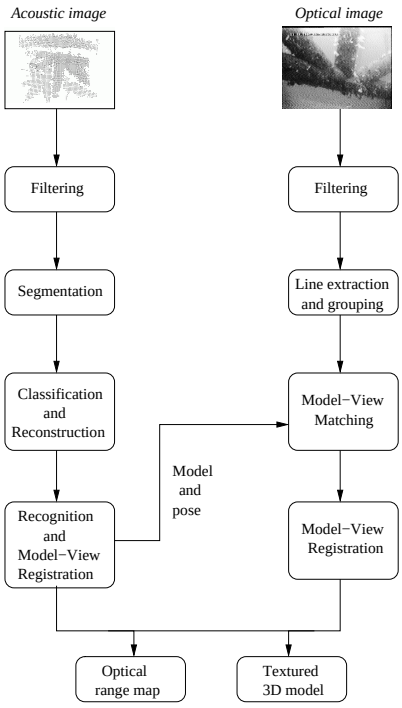


Figure 1: System overview. The single modules are described in the paper.

Accordingly, the optical data-processing thread is composed of the following modules:

- filtering, to reduce degrading noise while preserving edge information;
- line extraction and grouping, to estimate and combine straight contours to identify feature groups likely to belong to significant regions;
- model-view matching, by which the model and its pose estimated by the acoustic data processing phase are used to find the exact match between model features and image features to support the next registration phase;
- model-view registration, to register the model to the observed view and to estimate the relative pose.

Therefore, the poses of both sensors are computed with respect to the fixed observed object, and an actual registration and an integration of the two images are obtained. A range map of the optical image or a textured 3D model of the sensed object can now be shown to the operator, where the 3D information comes from the acoustic camera, whereas the texture comes from the video image.

Even though our approach considers a particular application domain and a specific class of objects, this does not limit its generality and usefulness in other contexts. On the other hand, general object-recognition systems are not yet completely automated, as they require the user's interaction, as in [14].

The overall process involves quite typical image-processing stages, which have been adapted and made more robust to manage the particular type of data, in particular, noisy and low-resolution acoustic data. More specifically, low/medium-level processing stages, like filtering, segmentation, and grouping (in terms of lines and regions) are shared by many vision tasks. Subsequent higher-level stages, like classification and reconstruction (for the acoustic channel) and recognition and model-view registration (for both channels), are obviously tailored to the kind of objects considered, but the adopted techniques are generally applicable to any object that can be decomposed into primitive elements.

3 Acoustic Sensing

In this section, we describe the processing of three-dimensional data obtained with the acoustic camera in order to register the sensed data to the model.

3.1 Acoustic Camera

Three-dimensional data are obtained with a high-resolution acoustic camera, the *Echoscope* 1600 [16]. The scene is insonified by a high-frequency acoustic pulse, and a two-dimensional array of transducers gathers the backscattered signals (see Fig 2). The whole set of raw signals is then processed (i.e., re-phased) in order to form computed signals (called *beam signals*) whose profiles depend on echoes coming from fixed steering directions, whereas those coming from other directions are attenuated. The distance of a 3-D point can then be measured by detecting the time instant at which the maximum peak occurs in the beam signal (see Fig 2). The 3-D image provided by the acoustic camera is formed by 64×64 points ordered according to a polar reference system, as adjacent points correspond to adjacent beam signals. Moreover, the intensity of the maximum peak can be used to generate another image representing the reliability of the associate 3-D measures, therefore, in general, the higher the intensity, the safer the associate distance.

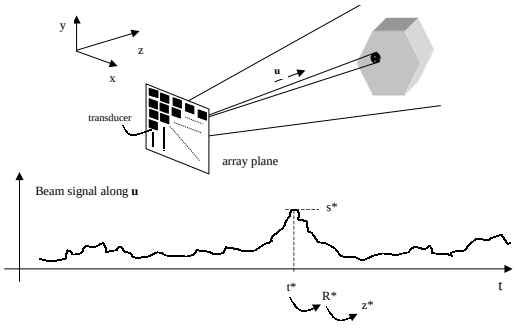


Figure 2: Functioning of the acoustic camera: after the insonification of the scene, backscattered echoes are acquired and processed to form *beam signals* coming from specific directions; the maximum peak of the beam signal identifies the scene distance in that direction.

3.2 Filtering and Segmentation

The acoustic image may be affected by false reflections, caused by secondary lobes, multiple returns, and acquisition noise, which is modeled as speckle noise. Although the Echoscope directly performs a preliminary low-level processing, it has proved useful in filtering 3-D data by a suitable algorithm. In particular, in the first step, connected components in the image are computed: two points are considered connected if they are adjacent in the 64×64 angular relation matrix and if their Euclidean distance is below a fixed threshold dependent on the spatial resolution of the camera. As a result, it is possible to subdivide the image into a certain number of connected components, while discarding those formed by a small number of points, that are not likely to represent interesting physical objects. In the second step, "reliable" connected components are formed by the points whose intensity is above a certain threshold, still dependent on the camera properties.

After this preprocessing phase, it is necessary to segment the image, i.e., to subdivide the set of 3-D points into distinct regions that are pipe candidates. To this end, the skeleton [17, 18] is first extracted and then used to subdivide the image into different convex components. The following procedure is applied to extract the skeleton: for every point $\mathbf{x}$, we consider all the points that are in a sphere of radius r centered on $\mathbf{x}$. Then, we shift $\mathbf{x}$ from its actual position to the centroid of such a distribution of points. The overall effect of this transformation is to shift points from the border toward the center, while leaving unaltered the points that are well inside the object considered. The iterative application of this procedure tends to cluster all the points of the distribution around the skeleton (see Fig. 3). Then,

the skeleton points are labeled as belonging to a branch or a joint by exploiting the properties of the inertial tensor (described in Sec. 3.3): for each point, the inertial tensor is computed in a small neighborhood, and the points showing a cylindrical symmetry are identified as branches. More details on this technique can be found in [19].

3.3 Classification and Geometric Reconstruction

The image segmentation by the skeleton extraction has provided us with a certain number of clusters of 3-D points that have been labeled as branches, which are the natural pipe candidates. They are now classified as pipe-like or non-pipe-like with a technique (related to the so-called Principal Component Analysis) based on the estimation of the *inertial tensor*.

Given a discrete distribution of N points $\{\mathbf{x}_i\}_{i=1\ldots N}$, the inertial tensor is the 3×3 matrix defined as

$$\mathbf{I} = \sum_i (\mathbf{x}_i - \mathbf{o}) \sqcap (\mathbf{x}_i - \mathbf{o}) \tag{1}$$

where $\mathbf{o}$ is the centroid and the symbol $\sqcap$ represents the following operator:

$$\mathbf{a} \sqcap \mathbf{b} = \begin{bmatrix} (a_y b_y + a_z b_z) & -a_x b_y & -a_x b_z \\ -a_y b_x & (a_x b_x + a_z b_z) & -a_y b_z \\ -a_z b_x & -a_z b_y & (a_x b_x + a_y b_y) \end{bmatrix}. \tag{2}$$

The eigenvalues and eigenvectors of $\mathbf{I}$ are then employed to extract useful information about the shape of the discrete distribution. Let $\alpha_1 \leq \alpha_2 \leq \alpha_3$ be the

eigenvalues of $\mathbf{I}$. If

$$\alpha_1 \ll \alpha_2 \text{ and } \alpha_1 \ll \alpha_3 \text{ and } \alpha_2 \simeq \alpha_3 \tag{3}$$

then the region is cylindrical in shape and is classified as a pipe; otherwise, it is discarded. To check on these relations, a threshold for the ratios α_2/α_1 and α_3/α_1 is introduced. The choice of this threshold is critical: if it is too low, it is probable to classify as a pipe something that is only elongate, whereas, if it is too high, it is possible that some pipes in the scene may be lost. A typical example of classification is shown in Fig. 4, where pipes are identified correctly.

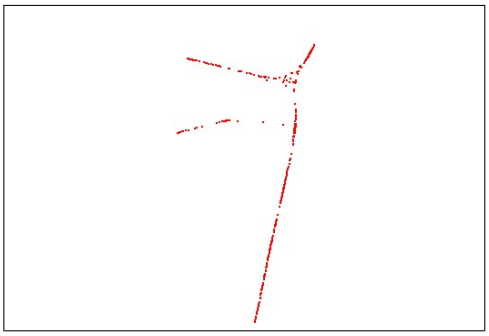


Figure 3: Skeletons extracted from data.

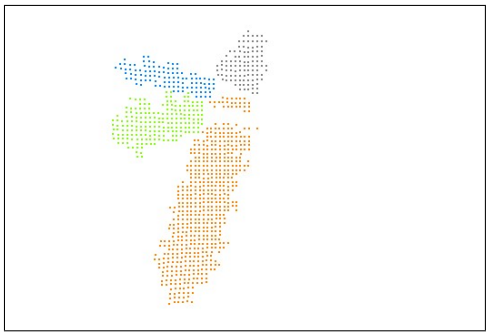


Figure 4: Segmented data. Each candidate pipe is in a different color.

From the value of the minimum eigenvalue it is possible to roughly estimate the radius of the tubular region. In the case of a complete cylindrical distribution, the following relation holds:

$$\alpha_1 = \frac{1}{2} N r^2 \tag{4}$$

where N is the total number of points in the distribution and r is the radius. Unfortunately, in acoustic images, points are not distributed on the whole surface of a cylinder but only on a little portion of it. Moreover, they are so noisy that they carry

little information about the curvature. Hence, relation (4) is only an approximation but is sufficient to give an order of magnitude for the radius, as will be seen in the section on the experimental results. In practice, we replace the term $\frac{1}{2}$ in Eq. (4) with an empirical constant k computed from synthetic images.

Finally, it is possible to determine the approximate position of the pipe axis (whose direction is given by the eigenvector relative to α_1) by translating the centroid of the distribution of r in the direction of the eigenvector corresponding to α_3, which is the radial direction.

In general, the axes of pipes belonging to a joint do not intersect exactly at one point or may not intersect at all. To extract an approximate intersection, we use the following simple algorithm: for every axis pair i, we compute the midpoint $\mathbf{m}_i$ of the single segment that connects the two lines defined by the axes and that is perpendicular to both of them.

If the number of axes is n, the number of possible pairs is $n(n-1)/2$. We define the center of the joint as the center of mass of the midpoints, i.e.,

$$\frac{\sum_{i=1}^{n(n-1)/2} \mathbf{m}_i}{n(n-1)/2}. \tag{5}$$

As we consider each *line* containing an axis, we retain only the intersections that are close enough to the axis endpoints.

This method works in a straightforward manner if there is only one joint in the scene; if this is not the case, it is first necessary to subdivide the set of extracted pipes into subsets containing pipes that belong to the same joint. To this end, it

is sufficient to group the pipes whose distance, defined as the distance between the lines passing through the axes, is below a threshold that depends on the radius of the pipes. This can be done by building the *incidence graph* G of the pipes, i.e., a graph whose nodes are the pipes and in which two nodes are connected if the distance between the corresponding pipes is below the given threshold. A joint corresponds to a maximal complete subgraph of G, i.e., a complete subgraph that is not contained in any larger complete subgraph. Two distinct joints can share no more than one node, corresponding to the pipe that connects them. The algorithm can be summarized as follows:

1. Start with the graph G of order n (the total number of pipes) and with an empty list of joints.

2. While $n > 1$, repeat the following steps:

3. Search for a complete subgraph of G of order n that is not contained in a subgraph of the list of joints.

4. If the subgraph exists, add it to the list of joints. Otherwise, decrement n.

A complete subgraph of order three may not represent a real joint but a triangle formed by three pipes. This is a degenerate case that is easily handled. It is sufficient to calculate the three midpoints $\mathbf{m}_i$ (defined above for the three pairs of pipes) and discard those whose distance is larger than a threshold.

For each of the remaining joints, the center is computed by using Eq. 5.

To sum up, the skeleton segmentation and the subsequent analysis with the inertial tensor are able to locate most of the pipes present in the observed scene and

to reconstruct, in a rough way, their geometrical properties. Although some pipes may be lost in this phase, a partial reconstruction is sufficient for the subsequent matching and alignment steps.

3.4 Recognition and Model-View Registration

After the extraction of the geometrical properties relevant to the joints in the observed scene, the model-object registration can be performed. Such properties are used to match these joints to the ones stored in the VRML model. In particular, we use the angles between the pipes as the recognition features: two joints match if such angles are equal within a certain error. As the joints analyzed are composed of a small number of pipes, the matching can be performed by an exhaustive method, although more sophisticated algorithms (e.g., based on Interpretation Trees [20]) can be adopted.

Acoustic data points lying on the surfaces of cylinders are matched to the underlying object surface model by using an iterative least-squares technique. Data points are expressed in the acoustic reference frame, whereas the model cylinders are placed in the model reference frame. The sought rigid transformation that links the two reference frames is given by $\mathbf{G}_a$, which is defined below.

In their paper, Besl and McKay [21] proposed the Iterative Closest Point (ICP) algorithm, a general-purpose method for the registration of rigid 3-D shapes. This approach eliminates the need to perform any feature extraction or to specify any feature correspondence.

3.4.1 The ICP algorithm

Suppose that we have two sets of 3-D points which correspond to a single shape but are expressed in different reference frames. We call one of these sets the *model set* X and the other the *data set* Y. Assume that, for each point in the data set, the corresponding point in the model set is known. The problem is to find a 3-D transformation that, when applied to the data set Y, minimizes the distance between the two point sets. The goal of this problem can be stated more formally as follows:

$$\min_{\mathbf{R},\mathbf{t}} \sum_{i=1}^{N} \|\mathbf{x}_i - (\mathbf{R}\mathbf{y}_i + \mathbf{t})\|^2, \tag{6}$$

where $\mathbf{G}_a = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$, $\mathbf{R}$ is a 3×3 rotation matrix, $\mathbf{t}$ is a 3×1 translation vector, and the subscript i refers to corresponding elements of the sets X and Y. Efficient, non-iterative solutions to this problem were compared in [22], and the one based on Singular Value Decomposition (SVD) was found to be the best.

The general 3-D registration problem that ICP addresses differs from the corresponding point-set registration problem in two important aspects. First, the point correspondence is unknown. Second, 3-D shapes to be registered are not necessarily represented as point sets.

Suppose that we have again two sets, X and Y, corresponding to a single shape, where Y is a set of 3-D points and X is a surface (of a cylinder, in our case). The correspondence between Y and X is unknown. For each point $\mathbf{y}_i$ in the set Y, there exists at least one point on the surface X that is closer to $\mathbf{y}_i$ than all the other points on X. This is the closest point, $\mathbf{x}_i$. The basic idea behind the ICP algorithm is that,

under certain conditions, the point correspondences provided by sets of closest points are reasonable approximations for the true point correspondences. Besl and McKay proved that, if the process of finding the closest-point sets and then solving equation (6) is iterated, the solution is guaranteed to converge to a local minimum. The ICP algorithm can now be stated:

1. For each point in Y, compute the closest point on X.

2. Using the correspondences from step 1, compute the incremental transformation $(\mathbf{R}, \mathbf{t})$ by SVD.

3. Apply the incremental transformation from step 2 to the data Y.

4. Compute the change in the total mean square error. If the change in the error is less than a threshold, terminate. Else go to step 1.

The ICP algorithm is only guaranteed to converge to a local minimum, and there is no guarantee that this local minimum will correspond to the actual global minimum. In our case, the recognition of the joint based on the estimated axes gives a fairly good initial alignment with the model, sufficient to achieve global convergence.

4 Optical Sensing

In this section, we describe the processing of the optical data in order to perform *registration*, that is, solving for the camera pose that best fits a model to some matching image features.

As the model is a tubular rig, the relevant image features are the segments forming the bounding contours of the pipes.[1]

4.1 Camera Model

The optical device is modeled by the *pinhole camera*, which is given by its *optical center* C and its *retinal plane* (or *image plane*) $\mathcal{R}$. A 3-D point W is projected onto an image point M given by the intersection of $\mathcal{R}$ with the line containing C and W (Fig. 5). The line containing C and orthogonal to $\mathcal{R}$ is called the *optical axis* and its intersection with $\mathcal{R}$ is the *principal point*. The distance between C and $\mathcal{R}$ is the *focal distance* (note that, as in this model C is behind $\mathcal{R}$, real cameras will have negative focal distances).

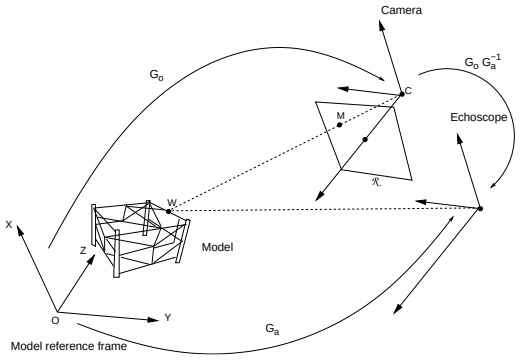


Figure 5: Optical/acoustic calibration.

Let $\mathbf{w} = [x\ y\ z]^\top$ be the coordinates of W in the *model reference frame*, and let $\mathbf{m} = [u\ v]^\top$ be the coordinates of M on the image plane (pixels). The mapping from 3-D coordinates to 2-D coordinates is the *perspective projection*, which is represented by a linear transformation in *homogeneous coordinates*. Let $\tilde{\mathbf{m}} = [u\ v\ 1]^\top$ and

[1] Given a viewpoint, the *rim* of an object is the set of all the points on the object surface to which the line joining the viewpoint (optical ray) is tangent (assuming perspective projection). The projection of the rim is the *bounding contour* of the object in the image.

$\tilde{\mathbf{w}} = [x\ y\ z\ 1]^\top$ be the homogeneous coordinates of M and W, respectively; then, the perspective transformation is given by the 3×4 matrix $\tilde{\mathbf{P}}$:

$$\lambda\tilde{\mathbf{m}} = \tilde{\mathbf{P}}\tilde{\mathbf{w}}, \tag{7}$$

where λ is an arbitrary scale factor. The camera is therefore modeled by its *perspective projection matrix* (henceforth PPM) $\tilde{\mathbf{P}}$, which can be decomposed, using the QR factorization, into the product

$$\tilde{\mathbf{P}} = \mathbf{A}[\mathbf{I}|\mathbf{0}]\mathbf{G}_o, \tag{8}$$

where $[\mathbf{I}|\mathbf{0}]$ is a 3×4 matrix composed of the identity and a column of 0's appended. The 3×3 matrix $\mathbf{A}$ depends on the *intrinsic parameters* only: focal length in pixels, aspect ratio, principal point, and skew factor. The camera position and orientation (pose) are encoded by the 4×4 matrix $\mathbf{G}_o$, representing the rigid transformation that brings the camera reference frame onto the model reference frame. $\mathbf{R}$ is the 3×3 rotation matrix and $\mathbf{t}$ is the 3×1 translation vector.

We seek the matrix $\mathbf{G}_o$, assuming that the *constant* intrinsic parameters have been computed off line by a calibration procedure [23].

4.2 Lines Grouping

Underwater images are characterized by a very low signal-to-noise ratio because of low illumination and bad environmental conditions. In order to filter the noise without affecting the signal, we use the Perona-Malik [24] anisotropic smoothing

filter, which preserves the information about object contours. Basically, it is a Gaussian smoothing filter with a standard deviation dependent on the grey-level gradient.

Straight lines are extracted by combining the Canny [25] edge detector with Burn's *Plane Fit Algorithm* [26]. First, edge points are extracted with the Canny edge detector, which allows one to find very sharp edges (often one-pixel large) thanks to the non-maxima suppression. Then, pixels are clustered into support regions if they are spatially adjacent and if their gradient orientations are roughly the same. The line parameters are computed with plane intersections of the weighted fit to the intensity values and the horizontal average pixel intensity plane, within a support region. The weight favours the intensity values of pixels with high gradient magnitude. Taking primarily the gradient orientation as evidence for a line and using the plane fit method, the algorithm actually extracts long, straight lines as well as shorter lines, and is effective in finding low-contrast lines.

Each extracted segment is then labeled and its attributes are computed. In order to find pipes in the image, pairs of segments are grouped together; they are likely to be the projections of the boundaries of a pipe (not every segment pair is the projection of a pipe). Grouping is based on proximity and *covering* criteria: two segments are paired if their projections onto their median axes overlap by more than 60%, and the distance between their midpoints is less than a threshold (which is related to the expected distance of the pipe boundaries in the image).

4.3 Model-View Registration

Optical alignment is performed using an algorithm (developed by Lowe [27]) that finds the camera pose that yields the best matching between each image segment and the projection of its corresponding cylinder rim. The algorithm assumes that image-model correspondences are given. In our case, the initial pose of the optical camera is assumed to be the same as that of the acoustic one ($\mathbf{G}_a$), already computed. Projecting the model accordingly, model segments are matched to the image segments by using an algorithm introduced by Scott and Longuet-Higgins [28] to associate the features of two arbitrary patterns.

If the approximate camera pose were unknown, a more complex recognition algorithm should be used [29].

4.3.1 The Scott and Longuet-Higgins algorithm

Scott and Longuet-Higgins [28] proposed an algorithm (based on the singular value decomposition (SVD)) for associating the features of two images. The algorithm incorporates both the principle of proximity and the principle of exclusion.

Let I and J be two images, containing m features I_i and n features J_j, respectively, which we want to put in one-to-one correspondence. The algorithm consists of three stages.

The first stage is to build a *proximity matrix* $\mathbf{G}$ of the two sets of features

$$G_{ij} = e^{-r_{ij}^2/2\sigma^2} \tag{9}$$

where r_{ij} is a well-defined distance between the features I_i and J_j, and σ is an

appropriate unit of distance that controls the scale of interaction. The next stage is to perform the SVD of $\mathbf{G}$

$$\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^{\top} \tag{10}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal and $\mathbf{S}$ is a non-negative $m \times n$ diagonal matrix.

Finally, $\mathbf{S}$ is converted into a new $m \times n$ matrix $\mathbf{D}$ by replacing every diagonal element S_{ii} with 1, thus obtaining another matrix

$$\mathbf{P} = \mathbf{U}\mathbf{D}\mathbf{V}^{\top} \tag{11}$$

of the same shape as the original proximity matrix and whose rows are mutually orthogonal. The element P_{ij} indicates the extent of pairing between the features I_i and J_j. If P_{ij} is both the largest element in its row and the largest element in its column, then we regard the two different features I_i and J_j as corresponding with each other.

This matrix incorporates the principle of proximity by construction of $\mathbf{G}$ and the principle of exclusion by virtue of its orthogonality.

In our application, the elements to be matched are lines, expressed in the normal form:

$$u \cos \alpha_i + v \sin \alpha_i - d_i = 0. \tag{12}$$

As a distance between model lines and image lines, we used the following

$$r_{ij} = \left\| \left[\cos \alpha_i, \sin \alpha_i, \frac{2d_i}{\max_l d_l} \right] - \left[\cos \alpha_j, \sin \alpha_j, \frac{2d_j}{\max_l d_l} \right] \right\| \tag{13}$$

The first two components are bounded in the interval $[-1, 1]$, whereas the third belongs to $[0, 2]$. As the initial pose is quite close to the true one, this simple matching is sufficient.

4.3.2 Lowe's algorithm

Let us suppose that *point correspondences* are available and that the intrinsic camera parameters are known. Let $\mathbf{w}_1 \dots \mathbf{w}_N$ be N points of an object model expressed in the model reference frame, and let $\mathbf{m}_1 \dots \mathbf{m}_N$ be the image points, projections of the $\mathbf{w}_i$. The relation between an object point and an image point is given by the perspective projection:

$$\kappa \mathbf{A}^{-1} \tilde{\mathbf{m}}_i = [\mathbf{R}|\mathbf{t}] \tilde{\mathbf{w}}_i. \tag{14}$$

derived from (8) by setting $\mathbf{G}_o = \begin{bmatrix} \mathbf{R}\ \mathbf{t} \\ \mathbf{0}\ 1 \end{bmatrix}$. Let $\tilde{\mathbf{p}}_i = [u_i, v_i, 1]^\top = \mathbf{A}^{-1} \tilde{\mathbf{m}}_i$ be the *normalized image coordinates*. If we expand, we see that each point correspondence generates two equations:

$$\begin{cases} u_i = \dfrac{\mathbf{r}_1^\top \mathbf{w}_i + t_1}{\mathbf{r}_3^\top \mathbf{w}_i + t_3} \\ v_i = \dfrac{\mathbf{r}_2^\top \mathbf{w}_i + t_2}{\mathbf{r}_3^\top \mathbf{w}_i + t_3}. \end{cases} \tag{15}$$

where $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]^\top$ and $\mathbf{t} = [t_1, t_2, t_3]^\top$. The 12 unknown components of $\mathbf{R}$ and $\mathbf{t}$ could be derived from a sufficient number of point correspondences by solving a linear system. The resulting $\mathbf{R}$, however, is not guaranteed to be orthogonal. To explicitly enforce its orthogonality, $\mathbf{R}$ must be parametrized with the three Euler angles ϕ, ψ, θ, ending up with a nonlinear system of six unknowns $\mathbf{t}, \phi, \psi, \theta$, which can be determined if at least three point correspondences are known. To counteract

the effect of inaccurate measures or correspondences, however, it is advisable to use as many correspondences as possible. The resulting overdetermined system of nonlinear equations can be solved (in the Least-Squares sense) by Gauss-Newton's method. This is usually referred to as Lowe's algorithm [27].

Equation (15) is linear with respect to translation and scaling over the image plane, and approximately linear over a wide range of values of the rotational parameters. Hence, the method is likely to converge to the desired solution for rather a wide range of possible starting positions. In our case, the initial pose of the optical camera is assumed to be the same as that of the acoustic one ($\mathbf{G}_a$), already computed. Given the small displacement between the two cameras and the negligible rotation, this is usually sufficient to ensure convergence.

The method can be easily extended to cope with *line correspondences* [27]. Given a set of pairs of corresponding image and model lines, we choose two points on each model line and compute the signed distance between each projected point and the corresponding image line. As each point gives one equation for the correction parameters, and as two points are sufficient to uniquely identify the model line, a line-to-line correspondence yields the same information (two equations) as a point-to-point correspondence, and the structure of the algorithm remains unchanged.

The case of smooth-boundary objects, like cylinders, is different. A rim generated by a sharp edge is stable on the object as long as the edge is visible, whereas a rim generated by a smooth surface changes continuously with the viewpoint. In our case, the rim is a line in space whose position is a function of the parameters $\mathbf{t}, \phi, \psi, \theta$. Hence, the expression for the residuals becomes more complicated. However, as

noted by Lowe [27], ignoring this dependence, hence treating the rim as a fixed line in space, does not prevent the algorithm from converging, and does not affect the precision of the final alignment.

5 Integration and Virtual Modeling

Given a rig composed of an optical and an acoustic camera, and given an acoustic image composed of a set of target points, each at a certain 3-D position, we want to project the acoustic image onto the optical image plane, thus obtaining a depth map with reference to the image plane.

To this end, the relative poses of the optical and acoustic cameras are needed. In principle, one should calibrate the cameras. A suitable object should be manufactured that is characterized by distinct features in both the acoustic and optical images. This is impractical underwater and very difficult to achieve, mainly because of the low resolution of the acoustic device. In our approach, we use the scene itself as a calibration object. Knowing the VRML model of the observed object, we register both the acoustic and optical data to the model, thereby obtaining the relative pose of the optical camera with respect to the acoustic cameras. As this process is performed on line, better estimates can be obtained by integrating the measures over time, using a Kalman filter [30].

Let $\mathbf{G}_o$ be the matrix representing the pose of the optical camera, obtained after the optical alignment, as described in Sec. 4:

$$\tilde{\mathbf{w}}_{\text{std}} = \mathbf{G}_o \tilde{\mathbf{w}}_{\text{model}}, \tag{16}$$

and let $\mathbf{G}_a$ be the rigid transformation that brings the acoustic-camera reference frame onto the model reference frame, computed by the 3-D alignment, as described in Sec. 3:

$$\tilde{\mathbf{w}}_{\text{sonar}} = \mathbf{G}_a \tilde{\mathbf{w}}_{\text{model}}. \tag{17}$$

By composing the two transformations, we get: $\tilde{\mathbf{w}}_{\text{std}} = \mathbf{G}_o \mathbf{G}_a^{-1} \tilde{\mathbf{w}}_{\text{sonar}}$. Hence, the PPM that projects the 3-D points expressed in the acoustic-camera reference frame onto the image plane of the optical camera is given by (see Fig. 5):

$$\tilde{\mathbf{P}}_{\text{oa}} = \mathbf{A}[\mathbf{I}|\mathbf{0}]\mathbf{G}_o \mathbf{G}_a^{-1}. \tag{18}$$

The intrinsic parameters' matrix $\mathbf{A}$ is the same as that of the optical camera, and is obtained by a calibration procedure.

By projecting the 3-D points onto the image plane, while keeping the third coordinate, which represents the distance of the points to the focal plane of the camera, we obtain a depth field defined at sparse locations. To obtain a proper depth map, a surface mesh is first generated by Delaunay triangulation on the image plane. The mesh may have several unwanted features upon creation, such as small, insignificant noise patches and jagged boundaries. Long edges and small unconnected surface patches are then removed. Moreover, as the acoustical data have been registered to the model, the points falling outside the pipe boundaries – because of the low spatial resolution of the acoustic device – are discarded. Finally, a uniformly sampled surface at a higher resolution than that of the original mesh data is obtained by interpolation and resampling the image over the pixel grid. As

a result, we achieve a depth map referred to the optical image.

Moreover, the accurate estimate of the position of the system relative to the environment is used in combination with the database information to provide a high-quality, 3D graphics, virtual display of the environment. This scene can be viewed from any position and direction, including the ROV itself, and as this virtual view is unaffected by turbidity, etc., it provides a clear and easily understandable view of the complete working environment.

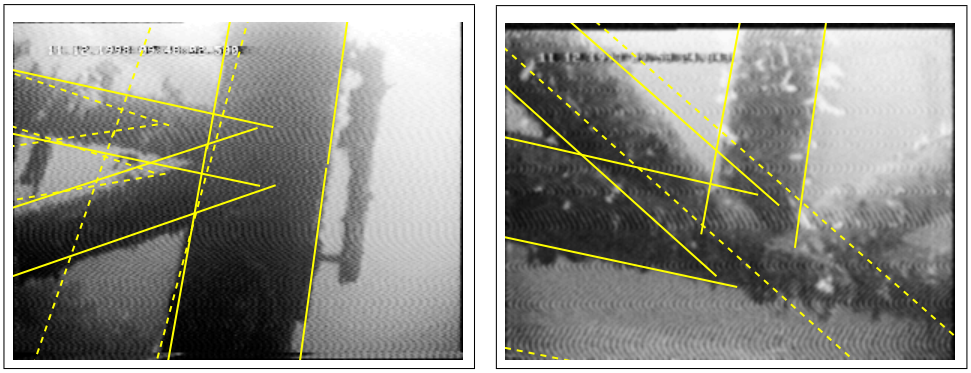
6 Results

(a) (b)

Figure 6: Raw acoustic data recorded with the acoustic camera. The joint is clearly visible, but there are also spurious points.

In this section, we provide the results obtained in two real cases. An ROV equipped with a video camera and an acoustic camera was used to take images of an underwater rig off Bergen, Norway. This rig constituted our model base and was approximately 20×20 m large and 10 m high; its VRML model was completely

known. The video camera was calibrated [23] underwater, using a suitable calibration jig in order to estimate the intrinsic parameters. The lateral displacement between the two cameras was approximately 300 mm, and the views were approximately parallel. However, we did not rely on these measures, for the relative pose of the cameras was obtained as explained in the previous sections.



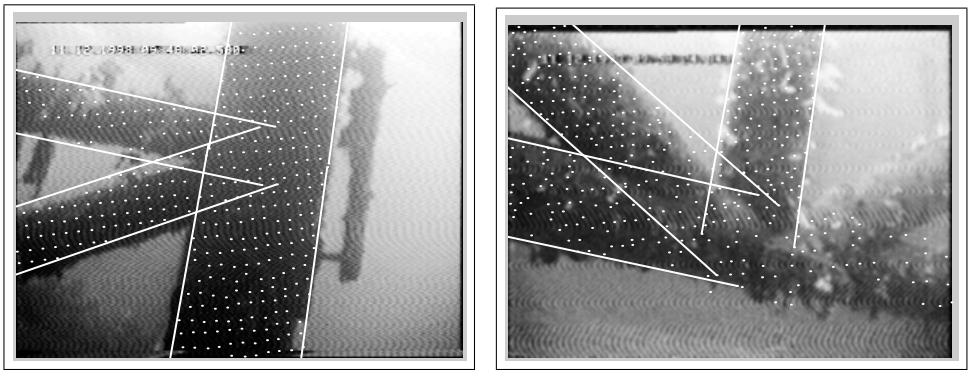
(a) (b)

Figure 7: Optical images with the projected model superimposed according to the initial pose estimate (dashed lines) and to the final pose estimate (solid lines).

Our procedure started from the raw acoustic data (in Fig. 6(a) and 6(b)) and the video image (visible in Figs. 7(a) and 7(b)) of a scene consisting of pipes of radii 500mm and 250mm meeting at a joint. These are typical cases used under operating conditions, as the view frustum of the acoustic camera cannot be too large due to interference phenomena. One can notice the quite bad qualities of both kinds of images, especially the low range resolution of the acoustic images.

A small annex attached to the vertical pipe is visible in Fig. 7(a). This structure is contained in the geometrical model, and if detected, can be used to validate the recognition of the main pipe structure. It cannot be confused with the main pipe,

even though the pipe-extraction module had detected it, thanks to the different size.

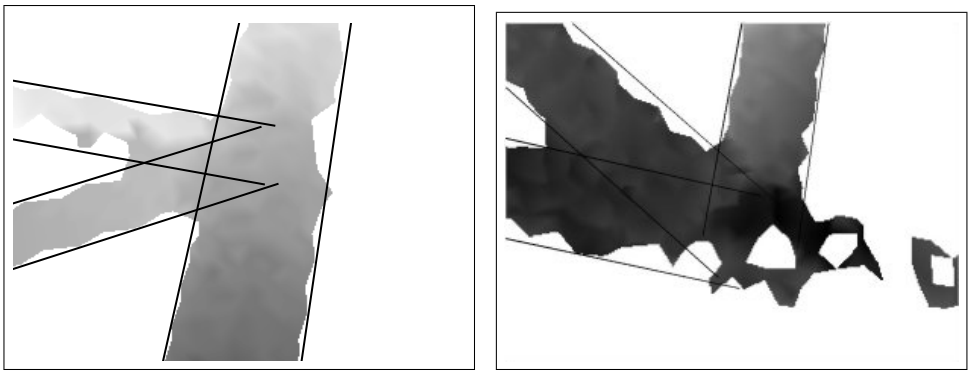


(a) (b)

Figure 8: Optical images with the projected acoustic points (white dots) and the projected model superimposed upon them. Points falling outside the pipe boundaries have been discarded.

When the vehicle is close to the structure, marine vegetation attached to the pipes may prevent them from being optically recognized (e.g., as in Fig. 7(b), where two pipes out of five have not been detected). However, the system still works because the matching to the model is guided by the acoustical data only.

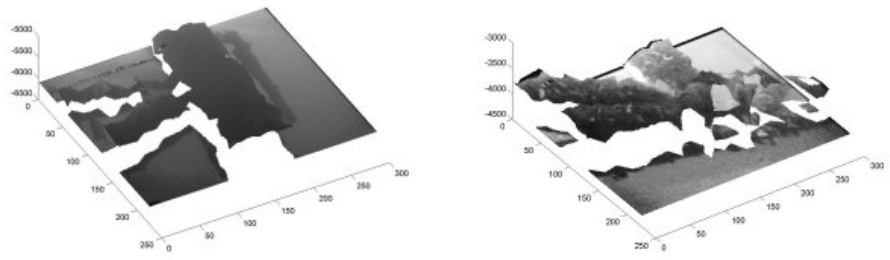
The viewing distance was 7.7m for the joint in Fig. 7(a) and 9.2m for the joint in Fig. 7(b). The registration of 3-D data converged to a solution with a residual (RMS point-model distance) of approximately 70 mm in both cases. This is a good result, as compared with the camera range resolution, which was 10 cm. The results of the optical registration are shown in Fig. 7(a) and Fig. 7(b), in both of which one can notice the accuracies of the optical model-view registrations starting from the rough alignment derived from the acoustic data processing phase. After both registration stages, the poses of the cameras with respect to the object were estimated and the acoustic-optical integration was actually carried out.



(a) (b)

Figure 9: Acoustic depth maps registered to the optical images. The gray level of a given pixel represents its depth (the darker, the closer). Solid lines represent the model projected according to the camera pose estimate.

The results of the integration can be appreciated in Fig. 9(a) and Fig. 9(b), which show depth images (registered to the optical images) where the depth for each pixel was computed from the projection of 3D acoustic points onto the optical images (see Figs. 8(a) and 8(b)).



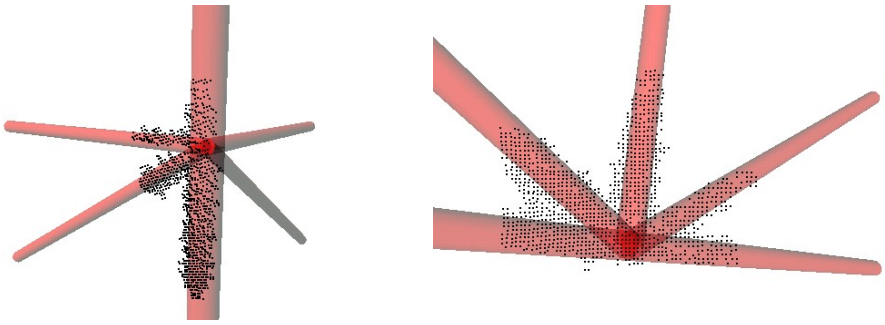
(a) (b)

Figure 10: Surfaces interpolating the (processed) 3-D acoustic points, with the real image texture mapped onto them. An arbitrary background plane is also shown.

In Fig. 10(a) and Fig. 10(b), the same depth maps are shown as surfaces, with

the original image texture-mapped onto them. Finally, Fig. 11(a) and Fig. 11(b) show the synthetic models with the 3-D acoustic points superimposed upon them.

The software prototype was written partly in C (GNU compiler) and partly in MATLAB on a Windows 95 Pentium 1 platform, and runs off line, taking as inputs one frame from the recorded sequence of acoustic data (in the form of x, y, z triplets) and the corresponding frame from the video sequence (in AVI format). Acoustic and video data carry time stamps (visible in the upper left corners of the images in Figs. 7 and 8). Calibration data are available in a configuration file. The overall computing time is about 10 seconds for the typical examples given in this section, the ICP algorithm being the most time-consuming operation, also because it is fully implemented in MATLAB. We are currently porting the system to Visual C++ on a Pentium IV 1.7 GHz computer, for the purpose of making it run at 10 frames per second.



(a) (b)

Figure 11: Virtual model of the scene with the 3-D acoustic points superimposed upon it. In the left picture the point of view was chosen such as to make visible two pipes that were occluded in the original image.

7 Conclusions

Guidance and inspection/maintenance/repair tasks performed by ROVs are very hard for several reasons. They require specialist crew, expensive training and many hours' practice. The output from the video camera is difficult to understand due to the 2-D nature of the images or to bad environmental conditions leading to disorientation. This situation cannot be significantly improved by using traditional acoustic sensors, as their outputs are not available in a form that is readily understandable even by a trained operator. The aim of our work is to overcome these difficulties.

This paper has presented a system aimed at assisting an ROV pilot by providing him with an augmented-reality image obtained by integrating multisensor data coming from an optical and an acoustic sensor and a VRML model. This virtual display of the working environment offers the basis for undertaking many typical underwater tasks with relative ease, as compared with current methods using video cameras only.

Available data are matched separately to a model in order to compute the pose of each sensor with respect to the model reference frame. In addition, the calibration of the two sensors leads to the registration of 3-D acoustic data to a 2-D optical image.

The system adopts some interesting methods for both acoustic and optical data processing. The most significant issues addressed by these methods are the synergic use of two different sensor devices, the calibration of the relative pose of the two sensors by using an observed object, and the integration of 3-D and 2-D data at the numerical level.

Acknowledgments

This work was supported by the European Commission within the framework of the BRITE-EURAM III project no. BE-2013 called VENICE[2] (Virtual Environment Interface by Sensor Integration for Inspection and Manipulation Control in Multifunctional Underwater Vehicles). The authors would like to thank Dr. R. Giannitrapani, who contributed to the developement of the acoustic data-processing, and Dr. R. Hansen of Omnitech A/S[3], Norway, for kindly providing the images acquired with the acoustic camera Echoscope 1600.

References

[1] Ronald T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and Virtual Environments*, 1997.

[2] M. Uenohara and T. Kanade, "Vision-based object registration for real-time image overlay," in *Computer Vision, Virtual Reality, and Robotics in Medicine '95*, 1995, pp. 13–22.

[3] J.P. Mellor, "Real-time camera calibration for enhanced reality visualization," in *Computer Vision, Virtual Reality, and Robotics in Medicine '95*, 1995, pp. 471–475.

[4] W.E.L. Grimson, T. Lozano-Perez, W.M. Wells, III, G.J. Ettinger, S.J. White, and R. Kikinis, "An automatic registration method for frameless stereotaxy,

[2] http://www.disi.unige.it/project/venice/
[3] http://www.omnitech.no

image guided surgery and enhanced reality visualization," in *CVPR94*, 1994, pp. 430–436.

[5] Ch. Schütz and H. Hügli, "Augmented reality using range images," in *Proceedings of SPIE Photonics West, The Engineering Reality of Virtual Reality*, San Josè, 1997.

[6] A. Johnson, P. Leger, R. Hoffman, M. Hebert, and J. Osborn, "3-D object modeling and recognition for telerobotic manipulation," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, August 1995, vol. 1, pp. 103 – 110.

[7] A. Johnson, R. Hoffman, J. Osborn, and M. Hebert, "A system for semi-automatic modeling of complex environments," in *Proceedings of the International Conference on 3D Digital Imaging and Modeling*, May 1997, pp. 213–220.

[8] F. Goulette, "Automatic CAD modeling of industrial pipes from range images," in *Proceedings of the International Conference on 3D Digital Imaging and Modeling*, May 1997, pp. 229–233.

[9] C.O. Jaynes, A.R. Hanson, E.M. Riseman, and H. Schultz, "Building reconstruction from optical and range images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1997, pp. 380–386.

[10] B. Gunsel, A.K. Jain, and E. Panayirci, "Reconstruction and boundary detection of range and intensity images using multiscale MRF representations," *CVGIP: Image Understanding*, vol. 63, no. 2, pp. 353–366, March 1996.

[11] G.H. Zhang and A. Wallace, "Physical modeling and combination of range and intensity edge data," *CVGIP: Image Understanding*, vol. 58, no. 2, pp. 191–220, September 1993.

[12] R. C. Luo and M. G. Kay, "Multisensor integration and fusion in intelligent systems," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, no. 5, pp. 901–931, September-October 1989.

[13] R.R. Brooks and S.S. Iyengar, *Multi-Sensor Fusion*, Prentice Hall, Upper Saddle River, USA, 1998.

[14] Y. Yu, A. Ferencz, and J. Malik, "Extracting objects from range and radiance images," *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, no. 4, pp. 351–364, October-December 2001.

[15] V. Murino and A. Trucco, "Three-dimensional image generation and processing in underwater acoustic vision," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1903–1946, December 2000.

[16] R. K. Hansen and P. A. Andersen, "A 3-D underwater acoustic camera - Properties and applications," in *Acoustical Imaging*, P.Tortoli and L.Masotti, Eds., pp. 607–611. Plenum Press, 1996.

[17] I. Pitas and A.N.Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Kluwer Academic Press, 1990.

[18] D. Attali and A. Montanvert, "Computing and simplifying 2D and 3D continuous skeletons," *Computer Vision and Image Understanding*, vol. 67, no. 3, pp. 261–273, 1997.

[19] V. Murino and R. Giannitrapani, "Three-dimensional skeleton extraction by point set contraction," in *Proceedings of the IEEE International Conference on Image Processing*, Kobe, Japan, October 1999, pp. 565–569.

[20] W.E.L. Grimson, T. Lozano-Perez, and D.P. Huttenlocher, *Object Recognition by Computer: The Role of Geometric Constraints*, MIT Press, 1990.

[21] P. Besl and N. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, February 1992.

[22] A. Lorusso, D. W. Eggert, and R. B. Fisher, "A comparison of four algorithms for estimating 3-D rigid transformations," *Machine Vision and Applications*, vol. 9, pp. 272–290, 1997.

[23] L Robert, "Camera calibration without feature extraction," *Computer Vision, Graphics, and Image Processing*, vol. 63, no. 2, pp. 314–325, March 1996.

[24] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[25] J.F. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, November 1986.

[26] J.B. Burns, A.R. Hanson, and E.M. Riseman, "Extracting straight lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 4, pp. 425–456, 1986.

[27] D.G. Lowe, "Fitting parameterized three-dimensional models to images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 5, pp. 441–450, May 1991.

[28] G. Scott and H. Longuet-Higgins, "An algorithm for associating the features of two images," in *Proceedings of the Royal Society of London B*, 1991, vol. 244, pp. 21–26.

[29] L. De Floriani, V. Murino, G.G. Pieroni, and E. Puppo, "Virtual environment generation by CAD-based methodology for underwater vehicle navigation," in *Signal Processing IX, Theories and Applications (EUSIPCO '98)*, 1998, vol. II, pp. 1105–1108.

[30] Arthur Gelb, Ed., *Applied Optimal Estimation*, The M.I.T. Press, 1974.

Andrea Fusiello received his Laurea (MSc) degree in Computer Science from the Università di Udine, Italy in 1994 and his PhD in Information Engineering from the Università di Trieste, Italy in 1999. He worked with the Computer Vision Group at IRST (Trento, Italy) in 1993-94, and with the Machine Vision Laboratory at the

Università di Udine from 1996 to 1998. He had been a Visiting Research Fellow in the Department of Computing and Electrical Engineering of Heriot-Watt University (UK) in 1999. As an Assistant Professor, he is now with the Dipartimento di Informatica, Università di Verona. He has published papers on image analysis, 3-D computer vision, and geometric reconstruction. His present research is focused on 3-D computer vision, autocalibration and image-based rendering. He is a member of the International Association for Pattern Recognition (IAPR) and IEEE. Further information can be found at http://www.sci.univr.it/~fusiello.

Vittorio Murino received the Laurea degree in Electronic Engineering in 1989 and the Ph.D. in Electronic Engineering and Computer Science in 1993, both at the University of Genova (Italy). He was a Post-Doctoral Fellow at the University of Genova, working in the Signal Processing and Understanding Group of the Dept. of Biophysical and Electronic Engineering. From 1995 to 1998, he was assistant professor at the Dept. of Mathematics and Computer Science of the University of Udine (Italy). He his now professor at the Department of Computer Science, University of Verona (Italy), and chairman of the same department. He worked at several national and European projects, especially in the context of the MAST (MArine Science and Technology) programme. His main research interests include: 3D computer vision and pattern recognition, acoustic and optical underwater vision, probabilistic techniques, data fusion, and neural networks with applications on surveillance, autonomous driving, visual inspection, and robotics. Recently, he is interested in the integration of image analysis and synthesis for object recognition and virtual reality modeling. Dr. Murino is author of more than 100 papers

in the above subjects, and associate editor of the Pattern Recognition and IEEE Transactions on Systems, Man, and Cybernetics journals, and the electronic journal ELCVIA (Electronic Letters on Computer Vision and Image Analysis). He is also referee for many international journals, and member of IAPR and senior menber of IEEE.