OTC 12001

Embedded Sonar & Video Processing for AUV Applications

D.M. Lane, E. Trucco. Ocean Systems Laboratory, Heriot-Watt University, Edinburgh

This paper was prepared for presentation at the 2000 Offshore Technology Conference held in Houston, Texas, 1–4 May 2000.

Abstract

We present recent work developing real-time sector scan sonar and video data processing algorithms for embedded use in AUVs and their topside data processors. Forward looking sonar algorithms have been designed and evaluated for object tracking, mapping, navigation and obstacle avoidance. Real time video algorithms have been evaluated for stabilization and docking on the vehicle, and for image stabilization, view synthesis, panoramic mosaicing and 3D shape reconstruction in post processing of data. Uncalibrated, or coarsely calibrated cameras are used throughout. The paper overviews key activities and developments, with references for more detailed descriptions.

Introduction

This paper reports ongoing work developing embedded sonar and video processing systems suitable for mounting on underwater systems such as AUVs. Applications are for vehicle navigation and obstacle avoidance, sea bed mapping and classification, visual servoing/docking, 3D shape reconstruction, and video image mosaicing.

For underwater vehicles to navigate safely and accurately in a possibly unknown environment, forward looking sonars can provide useful mission specific and navigation information. Examples are:

1- tracking of underwater objects
2- vehicle motion estimation
3- map building in a world reference frame
4- obstacle avoidance.

Similarly, real-time video processing can enhance vehicle utility, and be used to post process video data products for improved visualization and measurement. We have focused on the following:

1. Development of a *robust* tracker, which targets optimal features for video tracking and implements a completely automatic mechanism for rejecting unstable image features
2. A module recovering the camera/platform motion from the acquired video sequence (the homography).
3. Use of the motion data to stabilize the image viewpoint as the AUV swims adjacent to a target, or to synthesize alternative views of objects.
4. Real time panoramic mosaicing using clusters of such tracked points.
5. A module computing a 3D reconstruction of the viewed scene from the set of features declared reliable by the tracker in 2D images
6. Use of motion data to stabilize the vehicle relative to a structure in the field of view.

The former are useful tools for post processing AUV gathered video data for qualitative and quantitative analysis of subsea structures and seabed objects. The latter is relevant for AUV docking and accurate relative position keeping during inspection.

We consider sonar and video processing in turn.

Sonar Tracking and Navigation

We propose a system that builds a map of features in the environment, localises it-self with respect to these features, and avoids obstacles. The system uses standard *dead-reckoning* sensors and a mechanical forward looking sonar. Our previous research [1-3] presented different approaches for tracking returns from a forward looking sonar, for the purpose of classification and obstacle avoidance. Our current research tracks the position of objects in the environment and that of the vehicle itself, with respect to the world reference frame, to build a stochastic map [4] and allow the vehicle to navigate in this map whilst avoiding obstacles. The proposed system finds an *absolute position* estimate of the vehicle and the desired objects, which will be integrated with the vehicle's dead-reckoning sensors allowing for safe autonomous navigation.

Principles. The potential field approach used for avoiding obstacles has been widely used in the field of robotics for obstacle avoidance and path planning [5]. We review its basis, and the theory behind simultaneous localisation and mapping.

Artificial potential field. Path planning can be broadly divided into two classes of techniques: global and local. Global methodologies rely on the description of the obstacles in the configuration space of the vehicle while local techniques generally use *artificial potential field* or related techniques [6]. Global techniques consist of two main steps. The first step transforms the available description of the obstacles in the real world (Cartesian coordinates) to a suitable representation in the configuration space. The second step determines the free space of the configuration space and searches for the optimal path (with respect to a given criterion, i.e. time to goal, energy minimization...). These algorithms produce a collision free path if one exists. However, they are computationally intensive and require a priori knowledge of the environment.

Local techniques rely on the use of artificial potential functions and provide a fast and viable option to global techniques, especially for real-time applications and for applications where the environment can change rapidly or is only partially known. In local techniques, the obstacles are represented by a repulsive potential while the goal point is represented by an attractive potential. The super-imposition of the two potentials provides the desired workspace energy topology. It is possible to follow a path decreasing the potential to the goal. However, the workspace created can present local minima. A robust algorithm should be able to escape local minima while still avoiding obstacles.

The potential model. The potential field is represented as a discrete approximation of the analog field. This approximation is not a limitation as the data representing the objects is in general discrete and the discretisation process can be tailored to match the resolution needed by a given application.

The attractive field should have only one minimum point, the goal. A standard way of defining it is to use a parabolic well defined as:

$$U_{att}(q) = \tfrac{1}{2}\eta(q - q_{goal})^2 \qquad (1)$$

The force applied to the robot at any point q of the space driving it towards the goal is therefore:

$$F_{att}(q) = -\eta(q - q_{goal}) \qquad (2)$$

This form of potential has some useful properties as the force derived from it converges linearly to zero. Hence, the robot can be made to decelerate smoothly as it approaches the goal point.

The repulsive potential is defined as:

$$U_{rep}(q) = \begin{cases} \tfrac{1}{2}\iota\left(\frac{1}{\rho(q)} - \frac{1}{\rho_0}\right)^2 & \text{if } \rho(q) \le \rho_0 \\ 0 & \text{otherwise} \end{cases} \qquad (3)$$

It can be seen that the force exerted by the obstacle is zero when the robot configuration lies outside the distance of influence and tends to infinity as the robot approaches the obstacle preventing any collisions.

Planning algorithm. The classical technique uses standard gradient descent. Starting from the current robot configuration, the neighbour with the smallest potential amongst the ones having a smaller potential than the current configuration is selected. The process is iterated until the goal is reached or no neighbour can be selected. In the latter case, the current configuration is a local minimum.

In order to avoid the local minimum problem, we use a tree-searching algorithm that is complete. It is guaranteed to find a path in the free space if one exists. Again it operates by examining the neighbours of the current configuration, assuming that a path between adjacent configurations lies in the free space and that the configuration space lies in a bounded rectangular area.

A tree t of possible paths is built with the starting configuration at the root of the tree. At every iteration, the leaf of t with the lowest potential is selected. The neighbours of that configuration whose potentials are less than a given threshold (high in general but less than inside any object) and which are not already in the tree are selected. They are installed in the tree as successors of the current leaf. The algorithm terminates when the goal has been reached or any attainable configuration from the start has been installed in the tree. If the goal is been found, a path can be tracked back from the it to the start. Effectively it is the same as following the gradient descent technique until a local minimum has been found, in which case the algorithm acts by filling the well of the local minimum until a saddle point is found.

Simultaneous localisation and mapping. The first consistent solution to the simultaneous localisation and mapping problem was put forward by [4], and has lead to more recent work e.g. [7-9]. In [10] a thorough explanation on the underlying fundamentals is given. Overall, the basic approach is an augmentation on the extended Kalman filter [11-12], where the filter holds the relevant states of both the vehicle and the features used in the mapping and localisation process. The new state vector $\mathbf{x}(\cdot)$ assumes the following form,

$$\mathbf{x}(k) = [\mathbf{x}_v(k) \quad \mathbf{x}_1(k) \quad \mathbf{x}_2(k) \quad \cdots \quad \mathbf{x}_n(k)]^T \qquad (4)$$

The estimated error covariance (approximated mean-square error) for this system,

$$\mathbf{P}(k)=\begin{bmatrix}\mathbf{P}_{vv}(k) & \mathbf{P}_{v1}(k) & \cdots & \mathbf{P}_{vn}(k)\\ \mathbf{P}_{1v}(k) & \mathbf{P}_{11}(k) & \cdots & \mathbf{P}_{1n}(k)\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{P}_{nv}(k) & \mathbf{P}_{n1}(k) & \cdots & \mathbf{P}_{nn}(k)\end{bmatrix} \quad (5)$$

The state is propagated using the extended Kalman filter; i.e. assuming that the process and observation models are locally linear, and that the process and observation noise is small. The new state vector and associated covariance are thus,

$$\hat{\mathbf{x}}(k)=\mathbf{f}\left[\hat{\mathbf{x}}(k-1),\mathbf{u}(k),\mathbf{0},k\right] \quad (6)$$

$$\mathbf{P}(k)=\nabla\mathbf{f}_v\mathbf{P}(k)\nabla\mathbf{f}_v^{\;T}+\nabla\mathbf{f}_w\mathbf{Q}(k)\nabla\mathbf{f}_w^{\;T} \quad (7)$$

The filter can now be updated according to,

$$\hat{\mathbf{x}}(k+1)=\hat{\mathbf{x}}(k)+\mathbf{K}(k)\upsilon(k) \quad (8)$$

$$\mathbf{P}(k+1)=\mathbf{P}(k)-\mathbf{K}(k)\mathbf{H}(k)\mathbf{P}(k) \quad (9)$$

where

$$\upsilon(k)=\mathbf{z}(k)-\mathbf{h}[\hat{\mathbf{x}}(k+1\,|\,k),\mathbf{0},k] \quad (10)$$

$$\mathbf{S}(k)=\mathbf{H}(k)\mathbf{P}(k)\mathbf{H}^T(k)+\mathbf{R}(k) \quad (11)$$

$$\mathbf{K}(k)=\mathbf{P}(k)\mathbf{H}^{\mathbf{T}}(k)\mathbf{S}^{-1}(k) \quad (12)$$

For a vehicle using a sonar returning range and angle with respect to the vehicle frame the observation vector for a single feature will be $\mathbf{z}_i(k)=\begin{bmatrix}r & \theta\end{bmatrix}^T$, the full feature vector will be of the form,

$$\mathbf{z}(k)=\begin{bmatrix}\mathbf{z}_1(k) & \mathbf{z}_2(k) & \cdots & \mathbf{z}_n(k)\end{bmatrix}^T \quad (13)$$

Not all features will be observed by the vehicle at each iteration. Those that are observed must be associated to the features in the stochastic map state vector $\mathbf{x}(k)$. This is known as data association. Data association has been a subject of major research. A good description of various methods can be found in [12]. The simplest method is the gated nearest neighbour approach. This method is performed in innovation space and takes into account the uncertainty in both the sensor and the stochastic map state. This method performs well in clutter free environments. Observations that were not associated to an existing feature will be added to the stochastic map state and covariance. The new map state and associated covariance will be

$$\hat{\mathbf{x}}(k)\leftarrow\begin{bmatrix}\hat{\mathbf{x}}(k)\\ \hat{\mathbf{x}}_{n+1}(k)\end{bmatrix} \quad (14)$$

$$\begin{aligned}\mathbf{P}_{n+1n+1}(k) &= \mathbf{L}_{x_v}\mathbf{P}_{vv}(k)\mathbf{L}_{x_v}^{\;T}+\mathbf{L}_{z_{new}}\mathbf{R}(k)\mathbf{L}_{z_{new}}^{\;T}\\ \mathbf{P}_{n+1v} &= \mathbf{P}_{vn+1}^{\;T}(k)=\mathbf{L}_{x_v}\mathbf{P}_{vv}(k)\end{aligned} \quad (15)$$

Navigation. The aim is to develop a suitable algorithm which will than be easily integrated onto an existing AUV framework. The motivation for using a mechanically scanned sonar lies in its proven reliability, low cost and above all compact size and light weight. The Tritech DFS forward looking sonar presents itself as a versatile option and has been integrated into our framework.

Overview. The overview of the system can be seen in figure 1. The data from the sonar is segmented and the obstacles are fed into the stochastic map. The dead-reckoning sensors are used to improve the accuracy of the map. The obstacle avoidance algorithm is fed with a binary map where the regions that are occupied, and therefore must be avoided, are represented as ones and the empty regions are represented as zeros. The output from the obstacle avoidance algorithm is a path; this path is fed into the vehicle controller. The vehicle controller will in turn feed the stochastic map with the thrust values used at each iteration.

The Tritech DFS has two possible operating frequencies; the chosen frequency for the algorithm is 325 kHz. Which is a good compromise between resolution and speed. Typical working ranges will be 25 and a sector of size 60° will be scanned.

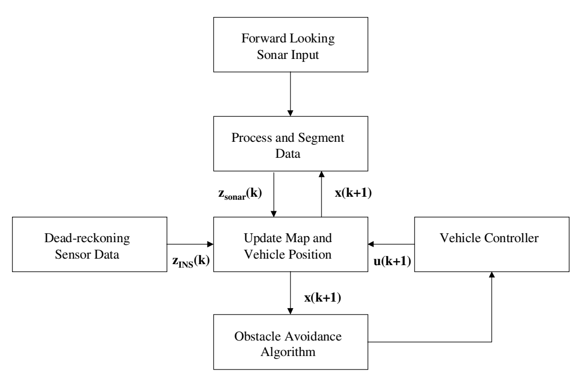

Fig 1 - Overview of the system

Data processing and segmentation. As the Tritech sonar has a 30° vertical beamwidth, in typical operation, 1 to 5 meters above the seabed, the seabed is in its field of view. The returns from the bottom should be removed to avoid high false alarm rates. In order to do that, the first significant returns are detected for each sonar beam and the bottom mean and variance is estimated. This estimate is continuously updated using a sliding window. If the parameters of the sonar are changed, the estimation is restarted from the first next image. A threshold is then derived from the mean and variance of the returns from the bottom, $\mu_{bottom} + 2 \times \sigma^2_{bottom}$. The image is then segmented using this threshold. If an object is in the part of the image where the bottom estimation is performed (close to the sonar), its effect will be minimal because of the sliding window. Once the image has been segmented the range and angle of the obstacles with respect to the vehicle is extracted and fed to the mapping and localisation module. Knowledge of the obstacles predicted position can also be used to relax the image processing constraints [3].

Dead-reckoning sensor data. The sensor suite mainly consists of off-the-shelf products. Currently these are:

- AOSI EZ-COMPASS-3: Tilt compensated magnetometer compass
- muRata Piezoelectric vibrating gyroscope: Single axis gyroscope

The data from these sensors is integrated and used to update the vehicle state in the Stochastic map. The system can benefit from the integration of better sensor technology., and to this end, the inclusion of further sensors is made easy under the Kalman filter structure.

Updating map and vehicle position. The stochastic map, holds the state of the vehicle and obstacles. A linear model of the vehicle dynamics is used, and has been found to work adequately. The state of the vehicle takes the following form,

$$\mathbf{x}_v(k) = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & \phi & \dot{\phi} & \ddot{\phi} \end{bmatrix}^T \quad (16)$$

Each obstacle has a state vector with states for its position and size,

$$\mathbf{x}_i(k) = \begin{bmatrix} x & y & s \end{bmatrix}^T \qquad (17)$$

The updating of the state is done as above.

Obstacle avoidance algorithm. All obstacles are stored in a global map referenced to a known starting position of the vehicle or any georeferenced position. The first step of the obstacle avoidance algorithm is to update the map of obstacles based on new data coming from the sonar. Once the data has been segmented as described above, each beam position is calculated using the time when the beam was collected and the stored timestamped positions of the vehicle during data collection.

The new data is then added to the global map. The new global map can now be used for path planning. If the positioning system of the vehicle is not accurate, the relative position between obstacles detected during different sonar scans will drift in time and eventually become meaningless. It is an option of the system to clean the map of objects detected in previous scans. It is even possible to use only the last scan when no positioning system is available.

The workspace is discretised using a given cell resolution. In each cell, the number of obstacles points is counted and if this number is above a threshold (corresponding to a number of points per square meter), the cell is considered as a obstacle and creates a repulsive potential around it. The path planning algorithm can thus be applied.

Results. The obstacle avoidance system has been successfully tested in water with a real sonar and vehicle dynamics (fig 2).

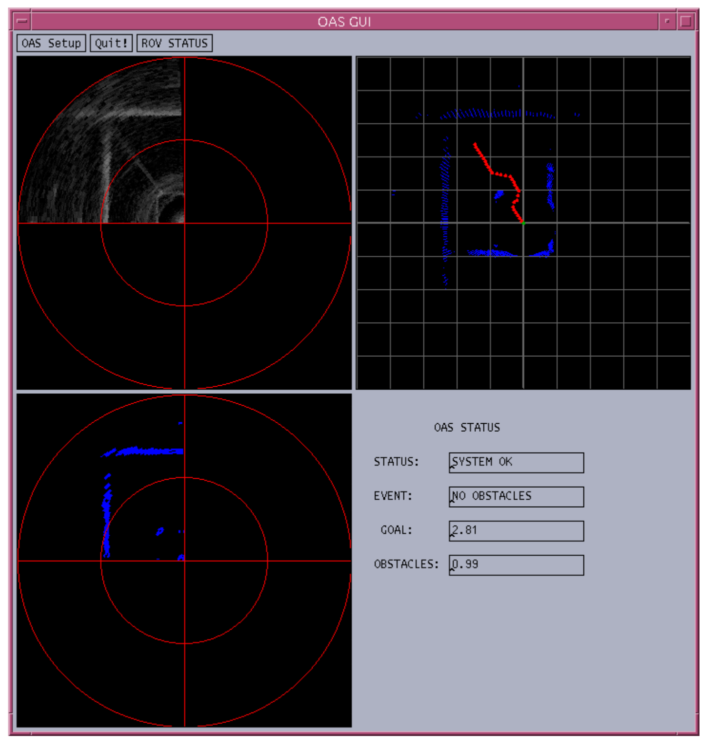

Fig 2 - Map of the estimated trajectory

Initially we used Ocean Systems Lab testbed ROVs and tank trials. Latterly we have used Ifremer VICTOR [14] and CNR-IAN ROMEO [13] UUVS within the EU DGXII ARAMIS benthic science toolskid development. We anticipate using Florida Atlantic University AUVs including Ocean Explorer and the new modular mini (figure 3).

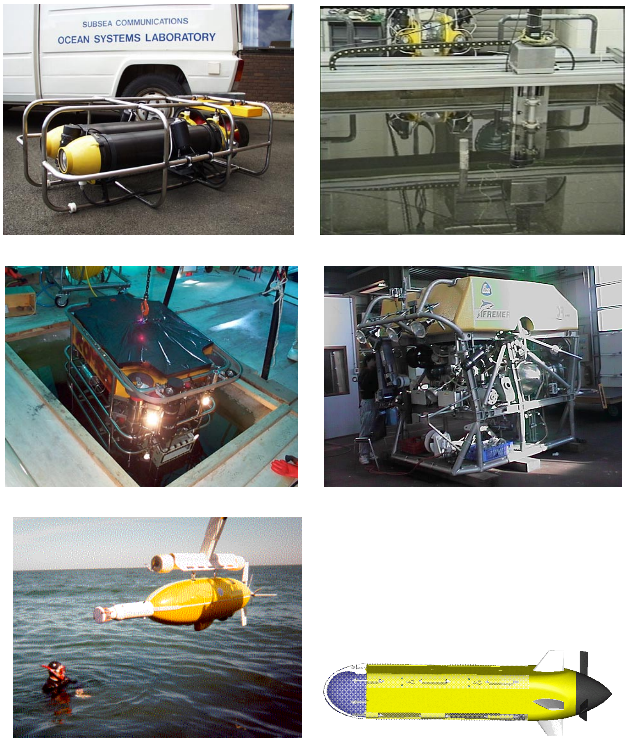

Fig 3: Testbed Platforms

Figure 4 shows the outcome of the algorithm for a simulated eighty-meter mission in which the vehicle used no dead-reckoning sensors. The measuring sensor had a 0.5 meter variance of gaussian distributed range error and a 1 degree variance of gaussian distributed bearing error. The average position error for the vehicle in x is of 0.50 meters, equivalent to 0.62% of the distance traveled, and the average position error for the vehicle in y is of 0.44 meters, equivalent to 0.55% of the distance traveled. Figure 5 illustrates the outcome for the same mission, the vehicle was now aided by a simulated AOSI EZ-COMPASS-3.

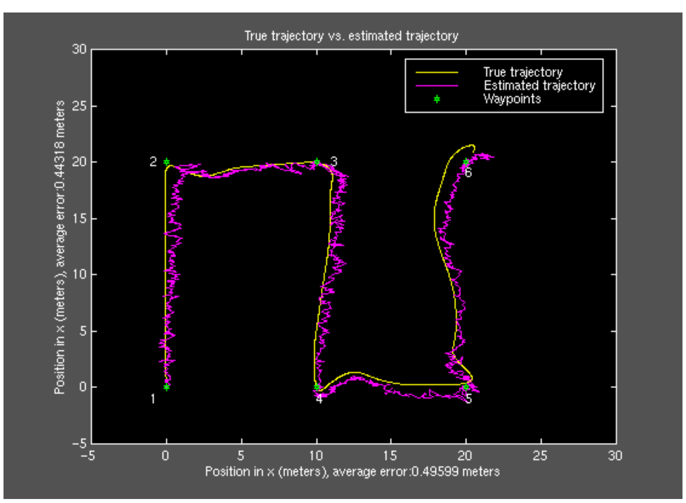

Fig 4 - Estimated trajectory vs. true trajectory using no dead-reckoning sensors

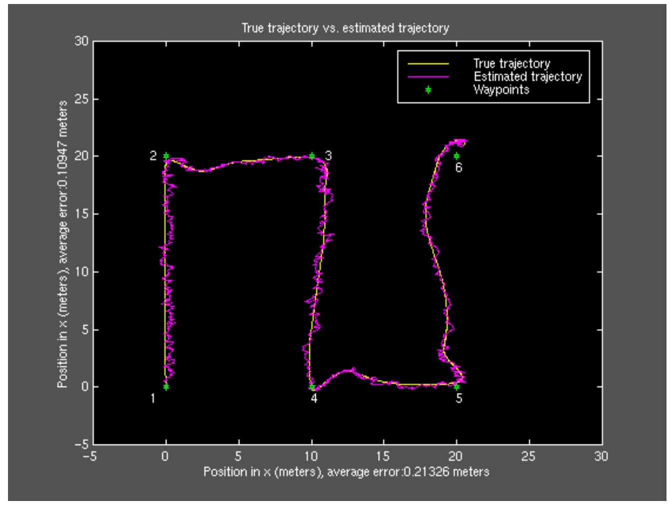

Fig 5 - Estimated trajectory vs. true trajectory using an AOSI EZ-COMPASS-3

The average position error for the vehicle in x is of 0.21 meters, equivalent to 0.26% of the distance traveled, and the average position error for the vehicle in y is of 0.11 meters, equivalent to 0.14% of the distance traveled.

Vision Processing

We have developed real-time embedded algorithms for image stabilization, view synthesis, panoramic mosaicing, 3D shape reconstruction, and vehicle stabilization. The former are useful tools for post processing AUV gathered video data for qualitative and quantitative analysis of subsea structures and seabed objects. The latter is relevant for AUV docking and video monitoring. Underlying these applications is a method for robust tracking of multiple image points, and computation of homographies (transformations) between frames. We digitize video and run these algorithms at frame rates using COTS video processing boards or fast Pentium processors.

Robust Video Tracking. Robust tracking means automatically detecting unreliable matches, or outliers, over an image sequence (see [15] for a survey of robust methods in computer vision, and [16-17] for trackers). We have developed a robust tracker based on an efficient outlier rejection scheme suitable for subsea video sequences (fig 6). The basis is the Shi-Tomasi-Kanade tracker [18-20], a feature tracker based on SSD matching and assuming affine frame-to-frame warping. The system tracks small image regions and classifies them as good (reliable) or bad (unreliable) according to the residual of the match between the associated image region in the first and current frames. Our work extends the Shi-Tomasi-Kanade tracker in several ways. *First*, we have introduced an automatic scheme for rejecting bad features, thus making the process fully automatic. We employ a simple, efficient, model-free outlier rejection rule, called X84, and have shown that its assumptions are satisfied in the feature tracking scenario. *Second*, we have developed a system running for indefinite lengths of time. *Third*, our tracker runs in real time on COTS hardware.

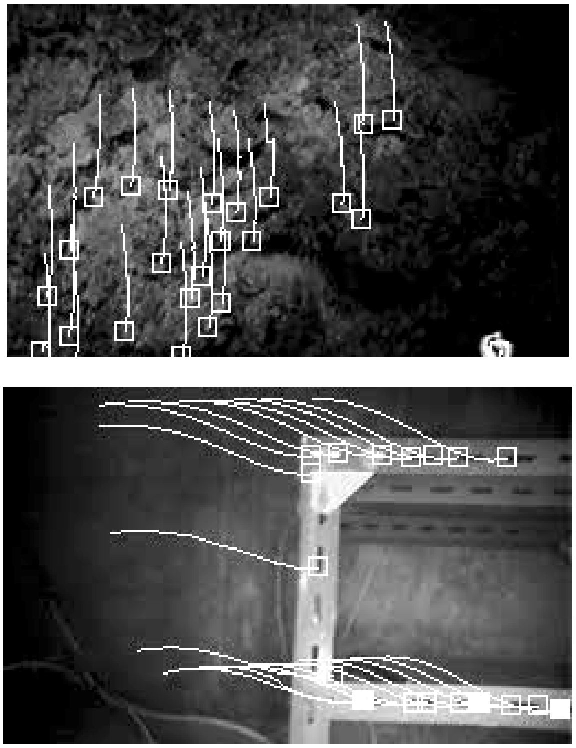

Fig 6 - Real-time tracking of multiple image points as a basis for shape reconstruction and relative motion estimation

Image Stabilization and View Synthesis. If we are able to track points through an image sequence from a moving platform, then we have the image point correspondences necessary to compute the *homography* (i.e. planar transformation) between the old and new projective plane (i.e. camera) locations.

An image is the intersection of a plane with a cone of rays between points in 3D space and the optical centre. Any two such images with the same optical centre will thus be related by a planar projective transformation. At least four point correspondences are required to define the transformation uniquely (within a scale factor). The linear solution is obtained by solving a set of 8 equations with 8 unknowns, or by using least squares or pseudo inverse methods.

With this homography identified, a synthetic rotation of the camera around its fixed optical centre can then be calculated. Image data can then be warped in real time, so that the image appears still during camera motion (figure 7), or alternative views of the same object can be obtained without moving the camera platform.

We have developed real-time software to implement this. which works well, even when underlying homography assumptions about co-planar objects are (weakly) violated.

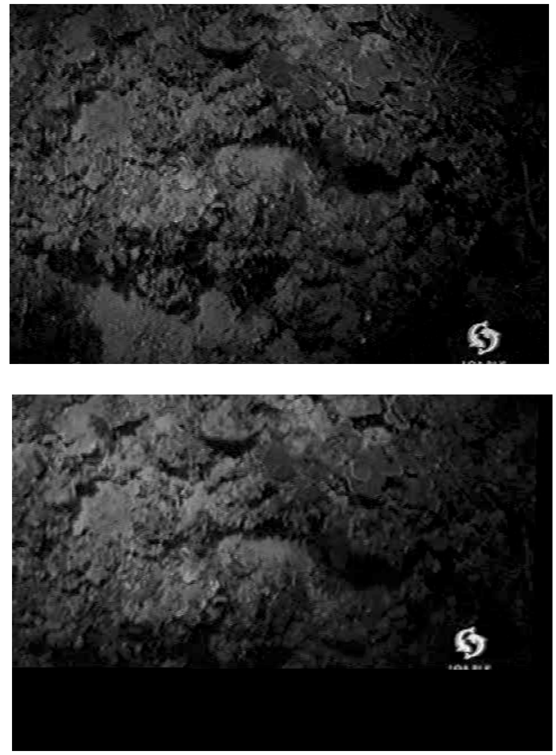

Fig 7 – Stabilized seabed views using view synthesis as AUV moves forward.

Panoramic Mosaics From Video Sequences. We have developed software for building panoramic mosaics [21-22] of scenes scanned by a video camera fitted on a vehicle (fig 8). No information is necessary on the camera motion; we assume that the scene is mostly planar, although experience indicates that good results (for panoramic displaying, not measuring) are achieved also when the scene is not entirely planar.

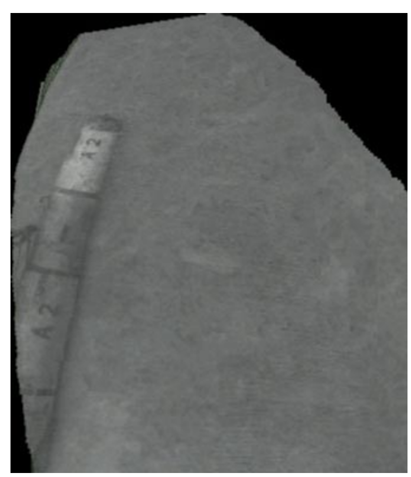

Fig 8 – Mosaic of subsea object from AUV mounted video

Reliable frame-to-frame correspondences are computed by the continuous video tracker (translational motion model) and fed into a module estimating the best plane homography aligning two consecutive frames [23-24]. The homography is then used to warp the current frame onto the reference one, that is, the current image is warped to simulate an image plane coplanar with that of the reference frame. The warped image can be aligned to the reference one without further geometric transformation. Image blending, guaranteeing seamless frame-to-frame borders, is implemented by locally averaging the values of all the pixels mapped to the same pixel in the final mosaic.

3D Structure From Uncalibrated Motion.

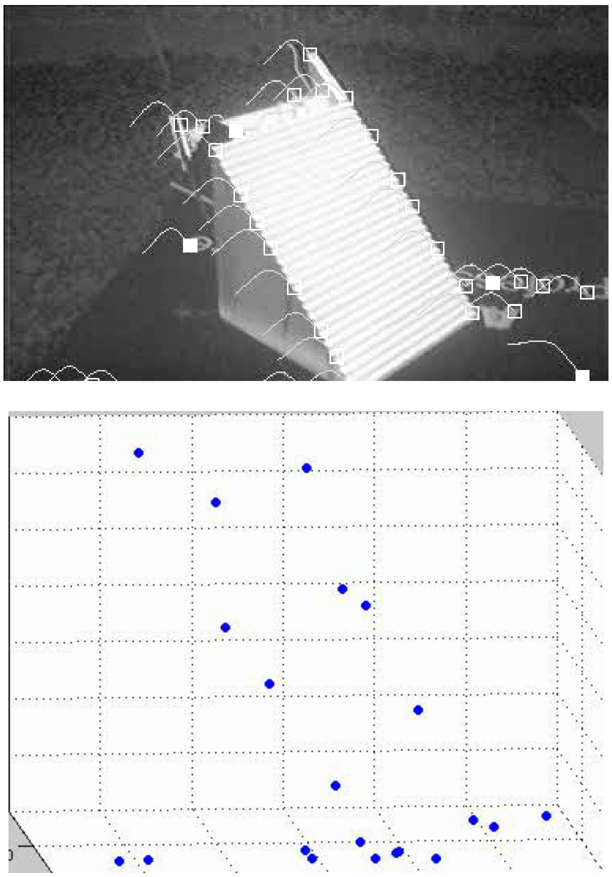

Fig 9 – 3D Reconstruction from image sequence. X,Y,Z of tracked image points are obtained from single uncalibrated camera, to within a scale factor

A video sequence acquired by a single camera in unknown motion contains complete information about the 3-D structure, but not the absolute size, of the scene observed [25-26]. The absolute size can be computed if the distance in space between two visible points is known. This means that video data acquired by AUV cameras can be used to build 3-D models of unknown targets (e.g., hydrothermal vents, rocks, wrecks) or to verify the shape and size of man-made structures (e.g., pipelines, valves, manifolds and similar installations).

An example of 3-D shape reconstruction from uncalibrated motion is shown in fig 9 [27]. The figure shows four frames from a sequence of a calibration pattern acquired in our laboratory tank by a camera moved by hand. No calibration at all was performed. The video tracker was used to compute reliable point correspondences throughout the sequence. The motion of the image points tracked were then used to estimate the shape (relative position) of the corresponding points in space. The reason for the small number of 3-D points reconstructed is that the feature set was never re-initialised. The accuracy of this reconstruction, computed as the average mean error in the relative position of known points, is approximately 2%.

Servoing and Vehicle Stabilization.

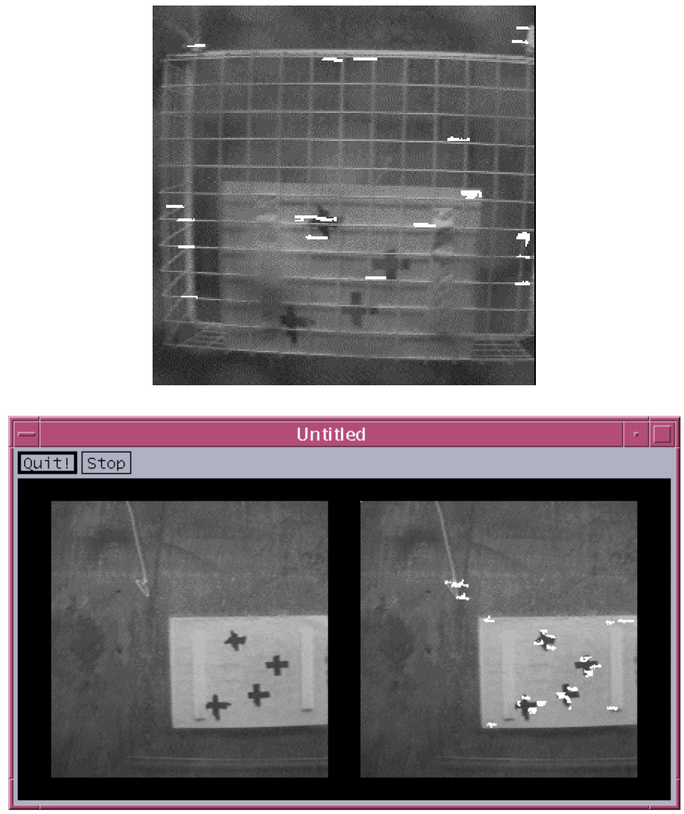

Fig 10 –Tracking during visual servoing using 2.5D motion estimation and bang-bang control. Note the small amount of hunting caused by the large inertia and low vehicle thrust in comparison to bang-bang controller expectation.

2D point tracking and homography estimation can also be used to stabilize the vehicle, rather than morph the image or estimate 3D structure. Object-relative navigation and control become feasible, provided optical visibility is adequate. Image based techniques attempt to directly identify the time varying, non-linear Jacobean matrix between changes in image and body axis co-ordinates. Position based schemes (such as here) explicitly recover the AUV pose for control.

We have experimented with use of the homography to derive UUV orientation and sign of position relative to the object in view (2.5D visual servoing), based on the original contribution in [28]. The method does not require information about 3D scene structure, and only coarse camera calibration. Orientation measurements are used with conventional PI controllers to stabilize the vehicle body angles. The sign-of-the-displacement measurements obtained from the 2.5D homography estimation are used with a bang-bang control scheme to stabilize in position.

Early results show that the bang-bang control introduces small amounts of hunting in the vehicle body axis translation outputs (figure 9). This is not unexpected, since good bang bang performance ideally requires zero AUV inertia and a large available thrust vector in the direction of motion, neither of which are generally available. We have found that close up (i.e. ranges less than 1m) surge velocities of 5cm/s are sufficient to lose tracker lock. This is because of the zooming effect, and the greater pixel motion for a given AUV motion at short range. This then removes the data association necessary to compute the planar transformation between camera (and hence AUV) locations, and servoing is no longer possible. The 2.5D homography estimation scheme can also be sensitive to noise in trackpoint identification. Good vehicle control and adequate target/lighting are therefore important during the final phases of docking stabilization with this visual servoing approach, to avoid loss of tracker lock.

Conclusions

The potential of the navigation algorithm has been demonstrated by the results. The obstacle avoidance algorithm is a promising tool that allows for a degree of AUV autonomy away from the open ocean. The stochastic map offers accurate position fixes of both the obstacles and the vehicle itself. It has been shown that these results can be expected to improve as further sensors and a more detailed vehicle model are integrated in the system. A limitation of the algorithm, which needs to be addressed, is the fact that the maps grow without bounds. Future work will consider the structure of these maps in closer detail. Optimal usage of maps by both the obstacle avoidance and mapping subsystems must also be addressed.

Embedded video processing algorithms have been described for improved visualization and 3D metrology in topside processing, and for servoing vehicle motion to stabilize relative to objects or seabed scenes containing sufficient trackable points.

Notations

q	a point in the workspace
η	scale factor
ι	positive scale factor
$\rho(q)$	smallest distance between the configuration of q and any obstacle in the workspace
ρ_0	positive constant defining the *distance of influence* of an obstacle
$\mathbf{x}$	stochastic map state vector
$\hat{\mathbf{x}}$	estimate of the stochastic map estate vector
$\mathbf{P}$	covariance matrix of the estimate of the stochastic map estate vector
$\mathbf{K}$	filter gain matrix
$\mathbf{S}$	filter innovation matrix
x_v	x coordinate of the vehicle
y_v	y coordinate of the vehicle
ϕ_v	orientation of the vehicle
$\mathbf{x}_v$	vehicle state vector
$\hat{\mathbf{x}}_v$	estimate of the vehicle state vector
$\mathbf{P}_{vv}$	covariance matrix of the estimate of the vehicle state vector
$\mathbf{x}_i$	feature i state vector
$\mathbf{P}_{ii}$	covariance matrix of the state estimate of feature i
$\mathbf{P}_{vi}$	covariance between the vehicle state estimate and the estimate of the state of feature i
$\mathbf{P}_{nm}$	covariance between the estimate of the state of feature n and the estimate of the state of feature m
$\mathbf{f}[\cdot]$	vehicle process model
$\mathbf{Q}$	process noise matrix
$\nabla\mathbf{f}_v$	Jacobian of vehicle process model with respect to the vehicle state
$\nabla\mathbf{f}_w$	Jacobian of vehicle process model with respect to the process noise
$\mathbf{h}[\cdot]$	observation model
$\mathbf{H}$	Jacobian of the observation model with respect to the state vector
$\mathbf{z}$	observations vector
$\mathbf{R}$	observation covariance matrix
r	range observation
θ	bearing observation
$\mathbf{L}_v$	Jacobian with respect to vehicle state of the transformation used to add a new feature to the state vector
$\mathbf{L}_z$	Jacobian with respect to the new observation of the transformation used to add a new feature to the state vector
υ	the innovation vector

Acknowledgments

This research is being supported by the European community under the MAST project CT97-0083 (ARAMIS). Significant contributions have been made by Yvan Petillot, Ioseba Tena Ruiz, Costas Plakas, Jean Francois Lots, Andrea Fusiello, Francesca Odone, Adriana Olmos, Anthony Doull, Katia Lebart and Alastair Cormack.

References

1. Lane D. M., Chantler M. and Dai D. Y. January 1998. Robust Tracking of Multiple Objects in Sector-scan Sonar Image Sequences Using Optical Flow Motion Estimation. IEEE Journal of Oceanic Engineering, 23(1):31-46.
2. Lane D. M., Chantler M., Dai D. Y. and Tena Ruiz I. April 1998. Tracking and Classification of Multiple Objects in Multi-beam Sector Scan Sonar Image Sequences. In Proceedings of the 1998 International Symposium on Underwater Technology, Tokyo, Japan, 269-273.
3. Petillot Y., Tena Ruiz I., Lane D. M., Wang Y., Trucco E. and Pican N. September 1998. Underwater Vehicle Path Planning Using a Multi-beam Forward Looking Sonar. In Proceedings of the IEEE OCEANS'98, Nice, France, 1194-1199.
4. Smith R., Self M. and Cheeseman P. 1990. Estimating Uncertain Spatial Relationships in Robotics. In Autonomous Robot Vehicles, Springer-Verlag, 167-193.
5. Latombe J. C. 1991. Robot Motion Planning. Kluwer Academic Publishers, Boston, USA.
6. Y. Wang and D. M. Lane. Subsea vehicle path planning using nonlinear programming and constructive solid geometry. IEE Proceedings on Control Theory Applications, 144:143--152, 1997.
7. Clark S. and Durrant-Whyte H. May 1998. Autonomous Land Vehicle Navigation Using Milimeter Wave Radar. In Proceedings of the 1998 IEEE International Conference on Robotics and Automation, Leuven, Belgium, 3697-3702.
8. Carpenter R., August 1998. Concurrent Mapping and Localization with FLS. In Proceedings of the 1998 Workshop on Autonomous Underwater Vehicles, Cambridge, USA, 133-148.
9. Feder H. J. S. 1999. Simultaneous Stochastic Mapping and Localization. PhD Thesis, Massachusets Institute of Technology, Cambridge, USA.
10. Csorba M. 1997. Simultaneous Localisation and Map Building. PhD Thesis, University of Oxford, Oxford, UK.
11. Maybeck P. S. 1982. Stochastic Models, Estimation, and Control, Volume 2. Volume 141 of the Mathematics in Science and Engineering, Academic Press.
12. Bar-Shalom Y. and Fortmann T. E. 1988. Tracking and Data Association. Volume 179 141 of the Mathematics in Science and Engineering, Academic Press.
13. Bono R., Bruzzone Ga., Bruzzone Gi., Caccia M., Spirandelli E. and Veruggio G. September 1998. ROMEO goes to Antartica. In Proceedings of the IEEE OCEANS'98, Nice, France, 1568-1572.
14. Cardiou J-F., Coudray S., Léon P. and Perrier M. September 1998. Control Architecture of a New Deep Scientific ROV: VICTOR 6000. In Proceedings of the IEEE OCEANS'98, Nice, France, 492-497.
15. P. Meer, D. Mintz, D. Y. Kim, and A. Rosenfeld. Robust regression methods in computer vision: a review. International Journal of Computer Vision, 6:59--70, 1991.
16. P. H. S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. In R. Fisher and E. Trucco, editors, British Machine Vision Conference, pages 655-664. BMVA, September 1996. Edinburgh.
17. P. H. S. Torr, A. Zisserman, and S. Maybank. Robust detection of degeneracy. In E. Grimson, editor, Proceedings of the IEEE International Conference on Computer Vision, pages 1037-1044. Springer--Verlag, 1995.
18. B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In Proceedings of International Joint Conference on Artificial Intelligence, 1981.
19. C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU- CS-91-132, Carnegie Mellon University, Pittsburg, PA, April 1991.
20. J. Shi and C. Tomasi. Good features to track. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 593--600, June 1994.
21. S. Peleg and J. Herman. Panoramic mosaics by manifold projection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 338--343, 1997.
22. A. Krishnan and N. Ahuja. Panoramic image acquisition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco (CA), 1996.
23. R. I. Hartley. Theory and practice of projective rectification. to appear in International Journal of Computer Vision, available from http://skipper.balltown.cma.com/hartley/, 1999.
24. F. Isgro and E. Trucco. Projective rectification without epipolar geometry. In Proc of the IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, June 1999.
25. E. Trucco and A. Verri. Introductory Techniques for 3-D Computer Vision. Prentice-Hall, 1998.
26. O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In Proceedings of the 2nd European Conference on Computer Vision, pages 563--578, S. Margherita(Italy), 1992.
27. K. Plakas, E. Trucco, and A. Fusiello. Uncalibrated vision for 3-d underwater applications. In Proceedings of OCEANS'98, volume 1, pages 272--276, Nice, September 1998. IEEE/OES
28. E. Malis, F. Chaumette, S. Boudet. Positioning of a course calibrated camera with respect to an unknown object by 2.5D Visual Servoing. Proc 1998 IEEE Int Conf, Robotics and Automation, Leuven, Belgium. pp 1352-1359.