A VISUAL SYSTEM FOR DISTRIBUTED MOSAICS USING AN AUVS FLEET

Silvia S. C.Botelho * **Paulo Drews** *
Marcelo Galeriano * **Eder Gonçalves** *

* *FURG Av. Itália km 8 Rio Grande, RS, Brazil*

Abstract: The use of teams of Autonomous Underwater Vehicles for visual inspection tasks is a promising robotic field. The images captured by different robots can be also to aid in the localization/navigation of the fleet. In a previous work, a distributed localization system was presented based on the use of Augmented States Kalman Filter through the visual maps obtained by the fleet. In this context, this paper details a system for on-line construction of visual maps and its use to aid the localization and navigation of the robots. Different aspects related to the capture, treatment and construction of mosaics by fleets of robots are presented. The developed system can be executed on-line on different robotic platforms. The paper is concluded with a series of tests and analyses aiming at to system validation.

Keywords: multi-AUVs, Visual Mosaic Systems, Localization

1. INTRODUCTION

Autonomous Underwater Vehicles (*AUVs*) can be applied to many tasks (Garcia *et al.*, 2005; Fleischer, 2000). In underwater visual inspection, the vehicles can be equipped with *down-looking cameras*, usually attached to the robot structure (Garcia, 2001). These cameras capture images from the deep of the ocean, supplying visual maps as the vehicle navigates. Each captured image is used to compose such map. Consecutive images are then aligned, generating a final map, also know as mosaic (Botelho *et al.*, 2005*a*). The generated mosaics can also be used as reference maps for the vehicle navigation system (Fleischer, 2000). This gives to the robot the ability to navigate in an autonomous way and in real-time. Given the robot altitude relative to the deep of the ocean (i.e. from the robot's altimeter) and the camera field of view, the covered area is known and real-time navigation is possible, using the described on-line construction.

In this context, we argue that simple robot fleets can be more efficient than a sophisticated AUV, in specific underwater exploration/inspection tasks. These simple vehicles could explore the same area in just a fraction of a single sophisticated AUV. More efficient mosaics can be generated and its visual information can be used to help the localization and navigation of the fleet.

In previous papers, an extension to the Augmented State Kalman Filter (ASKF) was presented, aiming the states estimation for a fleet of robots, assuming inspection robots with *down-looking cameras* (Botelho *et al.*, 2005*a*). In this paper a detailed Distributed Visual System for on-line mosaic construction is presented. The system supply information for the ASKF estimator. The images can be captured by simple low-cost underwater robots, associated to a underwater central station connected to the surface. The mosaic is computed by this central station.

Initially the paper presents related works on underwater mosaic construction. Section 3 presents a detailed view of our approach, followed by the implementation, test analysis and results. Finally, the conclusion of the study and future perspective is presented.

2. UTILIZING AUVS FOR VISUAL MAPPING

Vision system for mapping the deep of the ocean have been developed since the end of the 80's. In those systems the mosaic were used only as visual information for the users in the surface.

2.0.0.1. The construction of Visual Maps In order to assemble the mosaic, the various captured images (also known as frames) must be successively overlayed, resulting in a single visual map. Normally, the overlaying of those images takes a few steps.

Initially the images pass through a *Pre-processing* stage, when geometric deformations are corrected and informations inadequated for processing are removed. After that, the displacement between consecutive frames need to be measured. In the literature we can find a set of works in the frequency (Rzhanov *et al.*, 2000) or in the spacial domain (Olmos *et al.*, 2000). The latter can be *feature-based* (search by points of interest) or *featureless* (no need for points of interest). The relative movement between consecutive captured images can be estimated by *Homography* (planar matrix transformation) (Szeliski, 1994).

Having the displacement information, *Mosaic update* takes place. In this stage the determination of *when* and *how* a new image must be incorporated to the mosaic (temporal/spacial interval) occurs. Finally the *Image overlaying and mosaic construction* phase is executed. The pixels are combined so they overlay in the time-line (for example, average in time, medium in time, most recent pixel or least recent pixel).

2.0.0.2. Utilizing mosaics for AUV localization . The use of visual maps to assist in the localization of the vehicle introduces a new phase on the process: path estimation from the captured images, taking into account the need for on-line processing.

The previous works concerning utilization of mosaics for AUV localization consisted on detecting and correlating points in successive images. The Kalman Filter was used to predict point positions on the next image. The vehicle path was estimated using Generalized Hough Transform (GHT). Another approach created visual maps in real-time by using special-purpose hardware for image manipulation (Marks *et al.*, 1995). (Gracias *et al.*, 2002) uses the Harris and Stephes algorithm for the detection of border points as the points of interest, using first order derivatives of the images, estimating movement parameters (planar matrix transformation) for a sequence of images. A system for the generation of mosaics and real-time estimation of 3D displacement utilizing special-purpose hardware was developed by (Negahdaripour *et al.*, 1998).(Garcia *et al.*, 2005) applies texture operators and similarities measurement, as the Energy Filter, Co-occurrence Matrix, and others, making the correlation between consecutive images of the mosaic more precise. This approach was implemented directly in special-purpose hardware. The utilization of visual information to assist the dynamic stabilization of vehicles presented by (Perrier, 2005) should also be cited, as well as works in which the visual information is used in conjunction with informations acquired by other sensors (Kalyan *et al.*, 2005).

In the multi-AUVs context, issues associated with architecture and supervision (Spenneberg *et al.*, 2005), Mines Inspection using distributed sonar, new kinds of sensors, as *smart cables* (Yu and Ura, 2004) are presented in the literature. (Madhavan *et al.*, 2002) proposes the utilization of Kalman Filters for state estimation of wheeled mobile robots and known structured environments.

In this paper originally is proposed a fleet of AUVs, each one using visual information to improve on the localization task of the robots. The system employs ASKF for fleet state estimation (Botelho *et al.*, 2005*b*). Next section details a distributed visual system for the construction of visual maps by sets of AUVs.

3. A DISTRIBUTED VISUAL SYSTEM FOR THE CONSTRUCTION OF VISUAL MAPS

This work is based on the approach suggested by (Garcia *et al.*, 2005), extending it for distributed and on-line visual maps generation. Starting with pre-processing and manipulation of points correspondence, then using homographic techniques and finally the on-line assembly of the mosaic. The following subsections presents details of each one of these steps.

3.1 Pre-Processing and Detection of Points of Interest

The distortion caused by the camera lenses can be represented by a radial and tangential approximation. As the radial component causes a bigger distortion, most of the works developed so far corrects only this component (Gracias *et* *al.*, 2002; Garcia, 2001). Noises related to image acquisition are filtered out by using the Prewitt Filter (low-pass). Points of interest are obtained by a border detection algorithm (*corners*) (Shi and Tomasi, 1994). This technique utilizes first order derivatives of the image, resulting in high-gradient zones, as the equation 1:

$$M = \begin{bmatrix} \sum\limits_{S(p)} I_x^2 & \sum\limits_{S(p)} I_x I_y \\ \sum\limits_{S(p)} I_x I_y & \sum\limits_{S(p)} I_y^2 \end{bmatrix}, \tag{1}$$

where $I_x I_y$ is the point-to-point product, and $I_x = \left(\frac{\delta I}{\delta x}\right)$ and $I_y = \left(\frac{\delta I}{\delta y}\right)$ are approximations for the first order derivatives of the image's pixels in rows and columns. The matrix M represents the local auto-correlation function. For each p point the eigenvalues λ_1 and λ_2 of M are computed for the $S(p)$ neighborhood of $n \times n$ pixels dimension.

Candidate points that better represent the interest zones (borders) are searched. A good candidate point is one that has high eigenvalues. For each p point of the image only $\min(\lambda_1, \lambda_2)$ is considered. Following, a local maximum is computed for a $k \times k$ pixels neighborhood, rejecting all p points that satisfy equation 2:

$$\min(\lambda_1, \lambda_2) \leq q * \max(R(p)), \tag{2}$$

assuming $R(p)$ the $k \times k$ dimension matrix with $\min(\lambda_1, \lambda_2)$ as neighborhood values of p. The quality of the obtained borders points can be adjusted by the q parameter.

For the remaining points of interest, a minimum distance between them is certified, keeping the most meaningful extremities and discarding the others within a minimum distance.

3.2 Detection of Correspondence Region

To detect similar regions between consecutive images, the neighborhood associated with the interest points is analyzed, intending to compute similarity measurements in its neighborhood. Consider $n \times n$ correlation windows, centered at the m point of interest from the previous image and the $m\prime$ point from the current image [1] . Thus, for each pair $(m, m\prime)$ the Correlation Score (CS) is computed, selecting one which have the bigger CS for a given point m: $CS(m, m\prime) = \frac{\sum_{i=-\alpha}^{\alpha} \sum_{j=-\alpha}^{\alpha} I_m \cdot I\prime_m}{\alpha^2 \sqrt{\sigma^2(I) \cdot \sigma^2(I\prime)}}$ where $\alpha = ((n-1) \cdot q)/2$, $I_m = (I(x + i \cdot q, y + j \cdot q) - \overline{I(x,y)})$, $I\prime_m = (I\prime(x\prime + i \cdot q, y\prime + j \cdot q) - \overline{I\prime(x\prime, y\prime)})$, with I and $I\prime$ being the two subsequent images; $\sigma^2(I)$ is the variance of the correlation matrix and $\overline{I(x,y)}$ the mean of it. Only candidates with similarity degree greater than a threshold are considered. From those candidates, the one with bigger CS will be chosen. If there are no candidate with such a CS value, the point m does not present correlation between the current and the previous image.

3.2.0.3. Removal of False Correspondences To remove pairs $(m, m\prime)$ with CS values greater than the threshold but not valid correlations, a second criteria is verified. This criteria is the computation of the mean μ and the standard deviation σ from the Euclidean distances between all the $(m, m\prime)$ pairs considered as valid correlations. From these measures, all the pairs which distance is outside the $[\mu - t \cdot \sigma, \mu + t \cdot \sigma]$ are discarded, where t is the pertinence parameter. This parameter can vary depending from the nature of the movement and also the images to be correlated. Such removal stage returns a set of n possible valid correlation pairs. Spurious movements are eliminated with this procedure.

3.3 Estimating the Homographic Matrix

The images correlation provide a set of relative displacement vectors between the points associated to the found correspondence pairs. The n pairs are used to determinate the homographic matrix H. This homographic matrix will provide the estimated displacement between such images, transforming the homogeny coordinates into non-homogeny. The terms are operated in order to obtain a linear system, as the equation 3:

$$\begin{bmatrix} x_1\prime & 0 & \cdots & x_n\prime & 0 \\ y_1\prime & 0 & \cdots & y_n\prime & 0 \\ 1 & 0 & \cdots & 1 & 0 \\ 0 & x_1\prime & \cdots & 0 & x_n\prime \\ 0 & y_1\prime & \cdots & 0 & y_n\prime \\ 0 & 1 & \cdots & 0 & 1 \\ -x_1.x_1\prime & -y_1.x_1\prime & \cdots & x_n.x_n\prime & y_n.x_n\prime \\ -x_1.y_1\prime & -y_1.y_1\prime & \cdots & -x_n.y_n\prime & -y_n.y_n\prime \end{bmatrix}^T \cdot$$

$$\begin{bmatrix} h_{11} \; h_{12} \; h_{13} \; h_{21} \; h_{22} \; h_{23} \; h_{31} \; h_{32} \end{bmatrix}^T = \begin{bmatrix} x_1 \\ y_2 \\ \vdots \\ x_n \\ y_n \end{bmatrix} \tag{3}$$

3.4 Assembling the Mosaic

The mosaic construction concept of this paper allows a single vehicle to generate its mosaic and

[1] For performance reasons only pixels in q intervals inside the correlation window are analyzed, determining an effective analysis area of mxm pixels. Assuming $m = ((n - 1)/q) + 1$.

also multiple vehicles to construct it concurrently, using the same programming structure. The visual construction stages of the mosaic are detailed next.

3.4.0.4. Global Registry on the Mosaic Defined the projective transformation matrix between the previous image I and the current image I', knowing the average value of the distances between interest points m and m' from the images, respectively, is different from zero, it is possible to register globally the current image I', adding it and the H matrix to the mosaic structure, considering the H matrix is now referenced by ${}^{k}H_{k+1}$, where k references I and $(k+1)$ references I'. The global projective transformation [2] of the image I' on the mosaic can be defined by the equation 4.

$$ {}^{1}H_{k+1} = \prod_{i=1,\dots,k} {}^{i}H_{i+1}, \qquad (4) $$

3.4.0.5. The Distributed Construction Task To visually assemble the mosaic, the global projective transformation of each image belong to in the map must be known. As described by 4, ${}^{1}H_{k+1}$ represents this information and is used in $\tilde{m}^{1} = {}^{1}H_{k+1} \cdot \tilde{m}^{(K+1)}$, which provides where each pixel ($m^{(K+1)}$ position) from the mosaic's image $(k+1)$ must be written on the final mosaic ($\tilde{m}^{1}$ position). The points $\tilde{m}^{(K+1)}$ vary within $x \in [1 \dots l]$ and $y \in [1 \dots c]$, assuming l and c the captured image dimensions.

When multiple vehicles are executing exploration jobs, the matrix ${}^{1}H_{1}$ from the first image of each vehicle is defined by the equation 5:

$$ {}^{1}H_{1} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \qquad (5) $$

Notice that in this case each robot will be initially shifted by a position (t_x, t_y) from the inertial reference, so this position must be informed by the system operator on startup. Thus, considering that the matrix ${}^{1}H_{1}$ of each vehicle will be different, the homography ${}^{1}H_{k+1}$ of each one will provide the position on the final mosaic, therefore other transformations are not necessary.

3.4.0.6. Localizing the Fleet In previous works, the Augmented State Kalman Filter was extended to estimate the correct 3D position of the fleet and mosaic (Botelho *et al.*, 2005*a*). The ASKF is feeded with displacement informations obtained on the homography stage, correcting errors acquired throughout the process.

Capital cost of the system is limited to that of a camera, lighting, off-line storage and a capable processor (required for inspection tasks). No infrastructure such landmarks, beacons is necessary. However our system makes some assumptions: persistent appearance of the scene, majority flat 2D scene, restricted robot motion and initial position estimate of each robot.

The proposed Visual System was completely implemented. A set of tests are done, validating the proposal.

4. SYSTEM IMPLEMENTATION, TESTS AND RESULTS

The developed platform consist of a software system for each vehicle and another for the central station. The implementation followed a client-server architecture.The communication between the client and the server uses sockets, which utilizes TCP (Transmission Control Protocol). The client software is responsible for capturing the images obtained by the camera. Besides it process the correlation between those images, so the mosaic can be correctly assembled. The central station is responsible for gathering the information from all the vehicles, assembling the distributed visual map. A set of tests and analysis were executed to verify and validate the usage of the involved parameters and processes. Following there are some tests executed on desktop AMD Athlon 2800+ computers with 1Gb of RAM, both in the client and server side.

To validate the localization system an experiment was conducted, coupling a Panasonic camera model PV-GS15 to a robotic arm. The robotic arm consist of a *harmonic drive* actuator with a coupled encoder supplying angular readings each 0.000651 seconds. The data obtained by the encoder were compared with the information supplied by the vision system. The camera was pre-calibrated.

Test 1 Figure 1 shows the mosaic obtained by a rotation movement of the robotic arm around its own axis. The visual map is the result of rotation and translational movements of the camera coupled to the robotic arm. The generated mosaic was constructed with 200 captured and processed frames.

While the mosaic is being constructed, real-time position information is acquired. Figures 2(a) and (b) shows in green the displacement returned by the vision system and in blue the reference obtained by the coupled encoder, respectively.

The visual localization system follow the robotic system in a very satisfactory manner. Notice

[2] relative to the first reference image added

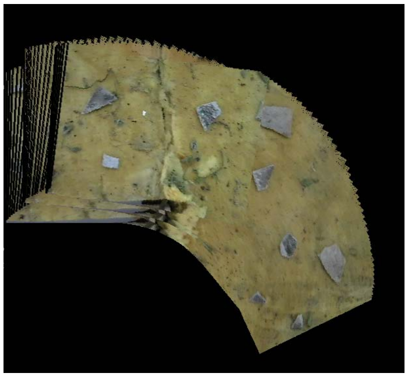

Fig. 1. Mosaic generated by the rotation and translational movement.

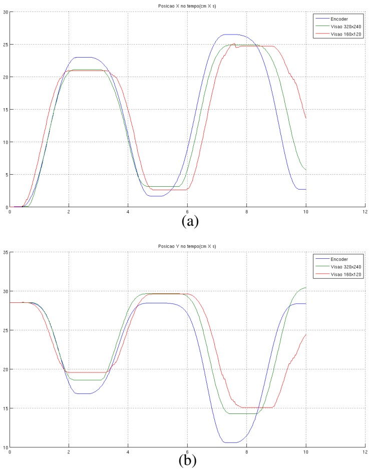

Fig. 2. Displacement in x (a) and y (b): mosaic generated by the rotation and translational movements - in green the encoder output, in blue the vision output

that no information related to position or pre-established landmarks are necessary.

Test 2 The same test was conducted with lower resolution images (half resolution). The results can be visualized in the figure 2, in red. Notice that with lower resolution images, in the case where the arm invert its movement (peaks on the curve), the small drift between the position returned by the vision and the reference is more evident. This is the result of the incapacity to detect the movement of the interest points by analyzing low resolution frames in smooth movement situations.

Test 3 The torque of the robotic arm was also varied in the same movement. Table 1 gives quadratic mean error associated with the position and the applied torque. Notice that the system has a appropriated behavior in a range between 0,261 N/m and 0,3828 N/m. In the extremed values, the results are worst due to the greater difficulty on obtaining the correlations between the interest points in small and fast velocities.

Fps	Num. Images	Correlation	Mosaic
4	359	140ms	52ms
3	253	138ms	53ms
2	112	136ms	53ms
1	41	134ms	53ms

Table 2. Mosaic construction times × Acquisition rate

Test 4 Figure 3 (a) and (b)shows in blue the speed of the arm, supplied by the encoder, and in green the speed obtained by the vision system. In this test the navigation time was considerable increased.

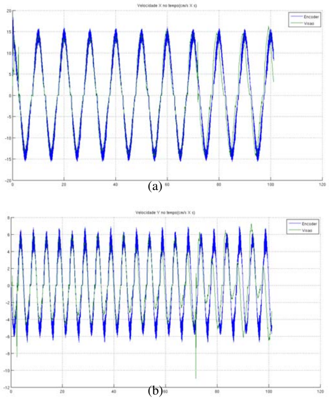

Fig. 3. Velocities in x (a) and y (b)- in blue the encoder output, in green the vision output

Test 5 The robustness of the system using various sampling rates was also verified. Table 2 presents the mean time for correlation and addition of frames to the mosaic. One can notice that these times do not vary significantly, and they make possible the use of the system in real-time applications.

Test 6 Tests were conducted with two robots connected to a central station. Figure 4 presents a visual map, generated in a distributed way with 195 images. Each robot was responsible for sending information to the central station at a three frames per second rate. The overlaying of images on the crossing region was satisfactory.

5. CONCLUSION

This paper presented an approach to the construction of visual maps using one or more AUVs. Theses maps might assist on fleet localization throughout a exploration mission.

Various tests were conducted, the effectiveness of the system was verified with different parameters as camera resolution, robot movement speed, mosaic quality and others. Results concerning processing performance was also presented. The correlation time was practicably constant and depends only on the nature of the obtained images. The mosaic construction time was also constant. Tests involving the multi-robot mosaic assembly

Torque Max	Num. Frames	Time (s)	X Error (cm/s)	Y Error (cm/s)	Angular Error (rad/s)
0.261 N/m	244	8.1	0.0715	1.2335	0.00130
0.3132 N/m	175	5.8	0.0399	0.3638	0.00048
0.348 N/m	146	4.9	0.0428	0.4718	0.00051
0.383 N/m	113	3.7	0.3095	0.6899	0.00120

Table 1. Quadratic mean error in the time associated with the position and the applied torque.

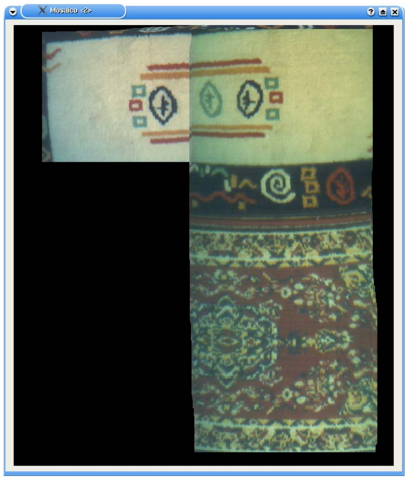

Fig. 4. Final Mosaic generated by 2 robots.

were conducted, resulting in good quality final mosaics, without on-line performance degradation.

Having finished this stage of mosaic distributed generation, for future works it is intended the integration with the ASKF system (currently running on MatLab) and AUVs robotic platforms, as well as the manipulation of stereoscopic images captured by stereo video heads. Also is intended tests submitting the system to underwater ambient characteristics like turbidity, sea snow, low illumination levels, and others.

REFERENCES

Botelho, S., R. Neves and L Taddei (2005*a*). Localization of a fleet of auvs using visual maps. In: *IEEE Oceans Conference*.

Botelho, S., R. Neves, L Taddei and V. Oliveira (2005*b*). Using augmented state kalman filter to localize multi autonomous underwater vehicles. *Journal of Brazilian Computer Science, to appear*.

Fleischer, S. (2000). Bounded-error vision-based of autonomous underwater vehicles. PhD thesis. Stanford University.

Garcia, R. (2001). A proposal to estimate the motion of an underwater vehicle through visual mosaicking. PhD thesis. Universitat de Girona.

Garcia, R., V. Lla and F. Charot (2005). Vlsi architecture for an underwater robot vision system. In: *IEEE Oceans Conference*. Vol. 1. pp. 674–679.

Gracias, N., S. Van der Zwaan, A. Bernardino and J. Santos-Vitor (2002). Results on underwater mosaic-based navigation. In: *IEEE Oceans Conference*. Vol. 3. pp. 1588–1594.

Kalyan, B., A. Balasuriya, H. Kondo, T. Maki and T. Ura (2005). Motion estimation and mapping by autonomous underwater vehicles in sea environments. In: *IEEE Oceans Conference*.

Madhavan, R., K. Fregene and L. Parker (2002). Distributed heterogeneous outdoor multi-robot localization. In: *ICRA02*.

Marks, R.L., S. M. Rock and M. J. Lee (1995). Real-time video mosaicking of the ocean floor. *IEEE Journal of Oceanic Engineering* **20**(3), 229–241.

Negahdaripour, S., X. Xu and A. Khamene (1998). A vision-system for real-time positioning, navigation and video mosaicing of sea floor imagery in the application of rovs / auvs. In: *4th IEEE Workshop on Applications of Computer Vision*. pp. 248–249.

Olmos, A., E. Trucco, K. Lebart and D. M. Lane (2000). Detecting ripple patterns in mission videos. In: *IEEE Oceans Conference*. pp. 331–335.

Perrier, M. (2005). The visual servoing system "cyclope" designed for dynamic stabilisation of auv and rov. In: *IEEE Oceans Conference*.

Rzhanov, Y., L. Linnett and R. Forbes (2000). Underwater video mosaicing for seabed mapping. In: *IEEE Conference on Image Processing*.

Shi, J. and C. Tomasi (1994). Good features to track. In: *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 593–600.

Spenneberg, D., C. Waldmann and R. Babb (2005). Exploration of underwater structures with cooperative heterogeneous robots. In: *IEEE Oceans Conference*. Vol. 2. pp. 782–786.

Szeliski, R. (1994). Image mosaicing for telereality applications. In: *IEEE Workshop on Applications of Computer Vision*. pp. 44–53.

Yu, S. and T. Ura (2004). A system of multi-auv interlinked with a smart cable for autonomous inspection of underwater structures. *International Journal of Offshore and Polar Engineering*.