Online Robust 3D Mapping Using Structure from Motion Cues

Tudor Nicosevici and Rafael Garcia

Computer Vision and Robotics Group
University of Girona, Spain
Email: {tudor, rafa}@eia.udg.edu

***Abstract*— This paper presents a complete solution for creating accurate 3D textured models from monocular video sequences. The methods are developed within the framework of sequential *Structure from Motion*, where a 3D model of the environment is maintained and updated as new visual information becomes available. The camera position is recovered by directly associating the 3D scene model with local image observations. Compared to standard structure from motion techniques, this approach decreases the error accumulation while increasing the robustness to scene occlusions and feature association failures.**

The obtained 3D information is used to generate high quality, composite visual maps of the scene (mosaics). The visual maps are used to create texture-mapped, realistic views of the scene.

I. Introduction

Visual mapping represents an important tool in underwater robotics with applications ranging from marine biology, geology, archeology to structural inspections. However, mapping in the underwater environment is inherently a complex problem. Light attenuation and backscattering drastically limit the range and coverage area of optical sensors; usually not more then a few meters. For this reason alone, large effort has to be devoted merely to align partially overlapping frames in order to generate a wider coverage what may be readily captured in a single frame in the absence of limited visibility. Most underwater mapping proposals found in the literature employ mosaicing techniques [1], [2] based on the assumption of a planar underwater terrain. However most of the areas of interest (eg. coral reefs, hydrothermal formations, ship wrecks, underwater structures, etc.) are hardly planar. In these cases the planarity assumption that characterizes homography-based approaches is violated, resulting in significant inaccuracies in the resulting visual maps. We propose a SFM-based method that enables 3D terrain modeling and localization of the camera in 6 degrees of freedom. The main novelty of the proposal consists in the estimation of the camera position without the need to recover the inter-frame motion as an intermediate step. A 3D model of the scene is maintained and updated as new data becomes available. The pose of the camera is computed directly from frame-to-scene correspondences. By directly estimating the camera position, rather than obtaining it by integrating sequences of motions, drift accumulation is highly reduced. Moreover, as each estimation of camera position does not depend directly on any previous estimations, it is more robust to misregistration problems and scene occlusions. The structure of the scene is formed by sets of 3D vertices characterized by local image descriptors. In this way, by visually associating image patches extracted from camera views with the 3D vertices, we can recover the camera position with respect to the scene model. Subsequently, the obtained camera positions are used to update the scene model as new features are tracked.

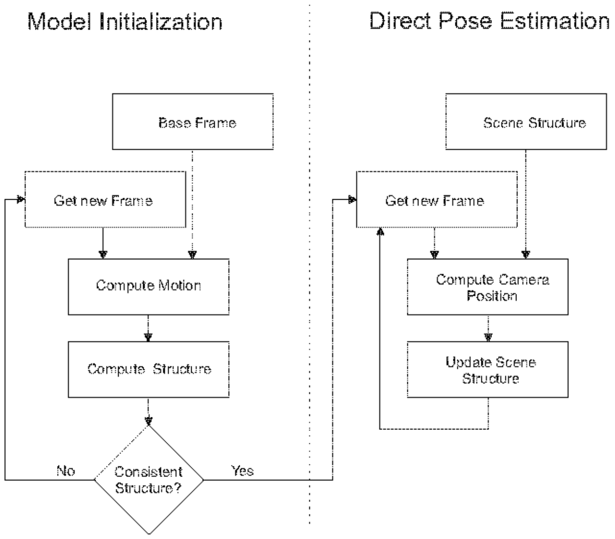


Fig. 1. Flowchart of the proposed SFM algorithm.

In the balance of the paper, we provide a detailed description of our system, present illustrative examples and experimental results, discussing some results obtained using real datasets, the efficiency of the proposal and its advantages. Finally, we conclude with some guidelines describing the further work.

II. Algorithm

The proposed SFM method computes directly the position of the camera without the necessity to recover the inter-frame motion (see Fig. 1). The structure of the scene is formed by sets of 3D vertices characterized by local image descriptors. In this way, by visually associating image patches extracted from camera views with the 3D vertices, we can recover the camera position with respect to the scene model. Subsequently, the obtained camera positions are used to update the scene model as new features are tracked. Both camera position estimation and scene model update use robust estimation methods thus

978-1-4244-2126-8/08/$25.00 ©2008 IEEE

reducing the impact of poor camera position/vertex estimations.

The proposed SFM algorithm works in two stages. First, it uses a standard motion estimation technique in order to initialize a model of the scene structure. Once an initial model has been generated, all the subsequent camera positions are computed by registering 2D point projections with 3D scene structure.

A. Feature Tracking

Feature tracking is the building block of any sparse 3D reconstruction algorithm. Robust feature tracking is crucial to the accurate estimation of both the camera positions and the structure of the scene. Maximizing the number of frames where a given scene feature is tracked increases the precision of its 3D position estimation and increases the number of inter-frame constraints, allowing a higher precision in camera position estimation. Nevertheless, tracking scene features in unstructured, cluttered scenes is not a trivial task. In the case of 3D scenes, especially at close range, the changes of view point due to camera motion induce quite severe distortions on local image patches (eg. rotation, scale, affine, etc.).

In the case of underwater scenes, there are additional aspects to be taken into account. In shallow waters, refracted sunlight can create flickering illumination patterns that change significantly from frame to frame. Additional artificial illumination used in deep water imagery induce non-uniform illumination fields [3]. Light scattering and absorbtion decrease the image contrast and induce blurring specially as the distance of the camera to the scene increases.

The proposed algorithm uses some of the most important feature detectors and descriptors found in the literature [4]. In order to assess the efficiency of these methods, we have tested them under various scenarios. Classical detectors such as, Harris [5], Hessian [6], Laplacian [7], provide fast detection but can only cope with small scale and rotation changes. Newer versions of these detectors called *Shape Adaptive* (SA) [8] such as Harris Affine and Hessian-Affine are more robust to geometrical distortions with some additional computational cost. The most robust and hence, widely used nowadays but also with the highest computational cost are Scale-invariant feature transform (SIFT) [9] and Speeded Up Robust Features (SURF) [10]. In contrast with the above-mentioned detectors that are essentially *interest point* detectors, we have obtained good tracking results using *Maximally Stable Extremal Regions* (MSER) detector [11].

The features are encoded using either SIFT [9] or SURF descriptors [10]. In the case of SIFT, each feature is described by a 128 elements vector embodying histograms of gradient directions within the feature patch. SURF and its variants (U-SURF, SURF-128, U-SURF-128, SURF-36, U-SURF-36) use 36, 64 or 128 dimension descriptor vectors characterizing the features in terms Haar wavelets responses on image patches.

The feature similarity is defined as the Euclidean distance between the descriptor vectors. Each descriptor vector can be seen as a noisy measurement of the image gradient within a feature patch. As the features are tracked, multiple measurements of the same patch are obtained. Hence, we can improve the measurement of the features by combining multiple observations:

$$s(F_k, f_k^i) = d(\frac{\sum f_k}{n}, f_k^i) \tag{1}$$

where $s(F_k, f_k^i)$ is the similarity between the tracked feature F_k and a possible observation (f_k^i) of the feature in image i and d is the Euclidean distance.

The feature tracking is done implicitly by continuously matching 2D image features with features corresponding to vertices in the 3D model. With the increase in model size this operation can become computationally expensive as the cost of matching is quadratic in the number of features. For this reason, we have employed an Approximate Nearest Neighbor search algorithm [12] that highly reduces the computational time associated to feature matching.

B. Model initialization

This step is used only for the initialization of the 3D structure of the scene, while no 3D information is available. The camera motion is obtained by Singular Value Decomposition of F into relative rotation and translation of the camera between frames [13], [1]. As this method is less accurate than direct camera position estimation, the purpose is to obtain an initial model in a reduced number of frames. For this, after testing various fundamental matrix estimation methods [14], RANSAC based Least Squared method has been adopted as it proved to provide more robust results in the case of small base lines. The camera motion can be obtained from F using [15]:

$$F = (K^{-1})^T \widehat{T} R K^{-1} \tag{2}$$

where K is the known camera intrinsic matrix, R is the rotation matrix of the camera and $\widehat{T}$ is the translation skew-symmetric matrix ($\widehat{T}_{[x]} = t \times x$ for any vector x) with t representing the camera translation. The approach yields 4 possible solutions (2 translations and 2 rotations). The correct solution is obtained by applying cheirality constraints (i.e. reconstructed points must be in front of the camera) [16].

Once the camera motion is obtained, the 3D model is estimated by applying classical stereo triangulation techniques. To obtain a consistent 3D model, an outlier rejection approach is used, based on Sampson distance [1] and robust sampling method (RANSAC).

C. Direct Pose Registration

We propose a method to directly compute the position of the camera by establishing correspondences between the camera view and the 3D structure of the scene. It does not use any a priori motion or position information. This method improves the robustness of navigation and mapping as it can naturally deal with occlusions, recovery from position estimation errors and loop closures.

The structure of the scene is represented by a set of 3D vertices in a common world frame (chosen as the coordinate

978-1-4244-2126-8/08/$25.00 ©2008 IEEE

system of the first camera position). Each 3D vertex has an associated descriptor vector (see section II-A). For each frame I_k we match the extracted image features with the 3D vertices representing the scene structure. The result is a set of 3D-to-frame correspondences that are used to determine the Maximum Likelihood estimate of the camera projection matrix P_k:

$$P_k = K \cdot (R_k|t_k); \tag{3}$$

Where R_k and t_k are the camera rotation and translation respectively at time k. P_k is obtained using Direct Linear Transform (DLT) and further adjusted using least squares methods in order to minimize the back projection error E_k:

$$E_k = \sum_i d(x_k^i, P_k X^i)^2 \tag{4}$$

Here, d represents the Euclidean distance, x_k^i is the projection of the feature i in frame k and X^i is the 3D position of feature i. In order to deal with the possible presence of outliers among the correspondences, a RANSAC-based approach is used. Once P_k is estimated, the position of the camera is recovered using eq. 3.

However, it is well known that if the scene points are close to being coplanar, the estimation of the projection matrix is ill-conditioned [1]. For dealing with this situation we propose a dual approach. For each RANSAC sample, we compute the plane L best fitting the 3D vertices sample set. If the mean vertex-to-plane distance is smaller than a given threshold, then we estimate the position of the camera using a homography-based approach. Here, for a given frame I_k we compute a planar transformation ${}^kH_L$ such as:

$$x_k^i =^k H_L \cdot x_L^i \tag{5}$$

Where x_L^i is the projection of X_i onto plane L. Applying SVD on ${}^kH_L$ [17], [18] we obtain:

$${}^kH_L =^k R_L + \frac{1}{d} \cdot^k t_L \cdot N^T \tag{6}$$

where ${}^kR_L$ and ${}^kt_L$ are the relative rotation and translation of between the plane L and the camera, N is the plane normal and d is a scaling factor. This decomposition yields 4 possible solutions. By checking the consistency of the camera motion with the scene structure (i.e. 3D points in front of camera) we can chose the correct transformation. From this, we obtain the pose of the camera such that:

$$\begin{aligned} t_k &= t_L \cdot^k R_L +^k t_L & (7) \\ R_k &= {}^kR_L \cdot R_L & (8) \end{aligned}$$

with t_L and R_L representing the pose of plane L in the world coordinate system.

As the camera moves, the SFM algorithm updates the model of scene as new features are extracted and tracked. Knowing the camera positions, the positions of the 3D vertices are obtained using a multi-view factorization approach [18]. The structure of the scene is continuously refined using least squares methods as new observations of the vertices are obtained (see eq. 4).

D. Ortho-mosaicing

The ortho-mosaicing process projects the 3D structure onto a plane, thus obtaining a virtual "high-altitude" view of the scene. The projection plane O is chosen to have the tilt as the average tilt of 3-D reconstructed surface. This maximizes the projection area, providing the highest level of mosaic detail. All the planar patches forming the 3-D model are mapped onto the destination plane along the projection rays parallel to the normal vector; see Fig. 2 as an example for the ortho-projection of a surface onto plane O.

The plane O is digitized based on a predefined resolution; each point x' on the grid corresponds to a pixel in the ortho-mosaic. In order to render the mosaic, the following transformation relating each point x' to a corresponding point x from the original images is defined:

$$x = P_k T_n x' \tag{9}$$

where T_n is the ortho-projection transformation of the patch $[X_1 X_2 X_3]$ and P_k is the camera projection matrix corresponding to the k-th view. The remaining problem is to determine which view I_k to use for rendering the point x'. In Fig. 2, the ideal image I_k to be used to render patch point x' would be the one minimizing the angle α between the surface normal $\overline{n}_X$ at point X and the projection ray $\overline{XF}_k$. This would provide the best projection of X into image I_k.

III. Experimental Results

The experiments were focused on testing the efficiency and precision of the proposed SFM algorithm. The proposal is intended as a mapping tool and a positioning system for robot navigation, we are interested in the precision of both the world model and camera pose estimation.

Here we discuss the results obtained using a dataset depicting a real underwater scene of a coral reef area. This dataset is part of a larger survey of a benthic habitat undertaken in shallow waters in The Bahamas. The images were acquired by the University of Miami using a hand-held High-Definition camera. The sequence consists of 1.100 images of 962 × 540 pixels (the resolution of the images was reduced from 1920 × 1080 due to interlacing). The area was surveyed with the camera moving following a "lawnmower" type trajectory (see Fig. 5), with partial overlap between adjacent columns. This provides a complete coverage of the area while offering additional constrains in the model.

The sequence was chosen to include different types of topologies and textures often found in underwater scenes. Fig. 3 depicts typical entities found in the dataset. The purpose of this was to assess the efficiency of the proposed algorithm to model these entities.

Prior to reconstruction, the camera was internally calibrated and the radial distortion was corrected. We have tested different types of feature extractors and descriptors. The highest

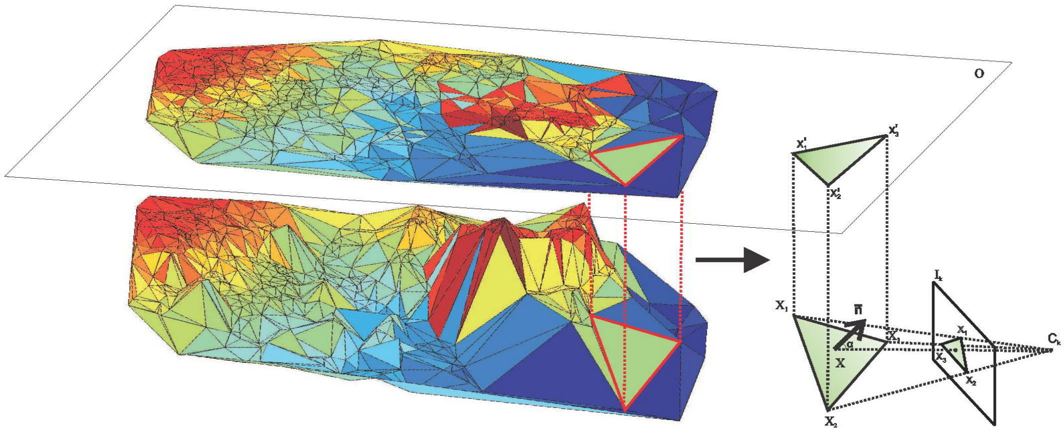


Fig. 2. Example of a 3D model and its Ortho-projection on plane O. Right part of the figure illustrates an example of texture-mapping of a triangle using camera view I_k.

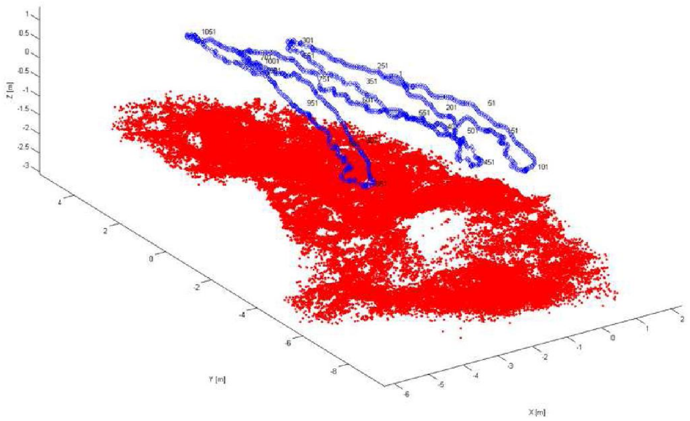


Fig. 4. Resulting set of 3D vertices and the camera trajectory. The regions with low vertex density correspond to high-slope scene surfaces with respect to camera viewpoint, offering little to no information for reconstruction.

stability and precision was obtained using Harris and SURF interest point detectors. The extracted features were characterized using SURF 64 dimension descriptors. After tracking and reconstruction, the resulting model consists of $\simeq 160K$ 3D vertices. Fig. 4 illustrates the set of 3D points obtained after the reconstruction along with the estimated camera trajectory.

As the experiment was carried out with real data, there is no ground truth available. This makes it difficult to assess the precision of the SFM algorithm. For this, we took advantage of the partial overlaps between non-consecutive frames in the sequence. By detecting these overlaps, we introduce new constrains in the model, and apply Bundle Adjustment (BA) techniques to obtain a high precision, globally consistent model. BA optimizes the camera pose and 3D vertex position estimates by minimizing a cost function defined by:

$$C = \sum_{ij} d(\widehat{P}^i \widehat{\mathbf{X}}_j, \mathbf{x}_j^i)^2 \qquad (10)$$

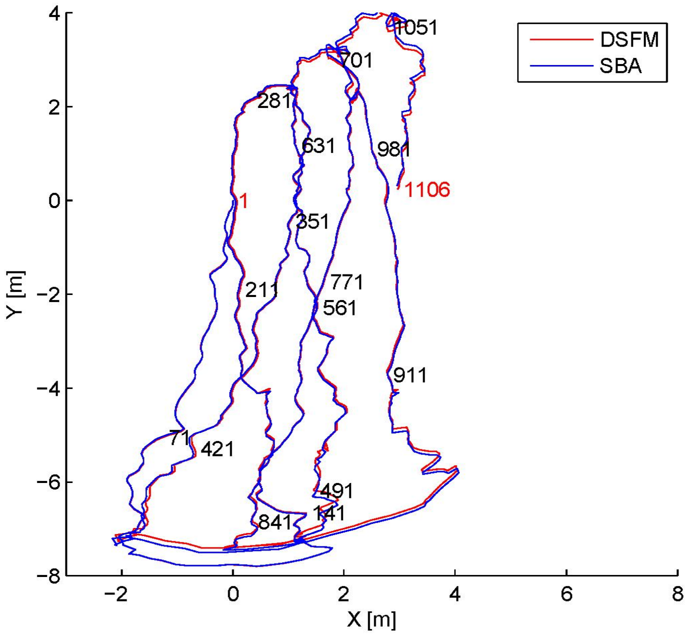


Fig. 5. Camera trajectory estimated by the DSFM and after global alignment

where C represents the back-projection error and d is the Euclidean distance [1]. C is minimized by adjusting all the camera positions $\widehat{P}^i$ and vertex positions $\widehat{\mathbf{X}}_j$.

Fig. 5 provides a comparison between the camera trajectory as obtained from SFM and the trajectory resulting from BA. Small errors in the estimation of camera position can be noted. However, unlike classical SFM approaches, the error build-up is highly reduced as the camera views are directly registered to the model. This can be better observed by analyzing Fig. 6. For instance, there is a slight drift in camera position estimation between frames 400 and 500. However, around frame 500, the errors are greatly reduced due to the registration of the camera with a region of the model that has lower errors (corresponding to frame $\simeq 160$).

Errors in the estimation of both camera poses and 3D vertex

978-1-4244-2126-8/08/$25.00 ©2008 IEEE

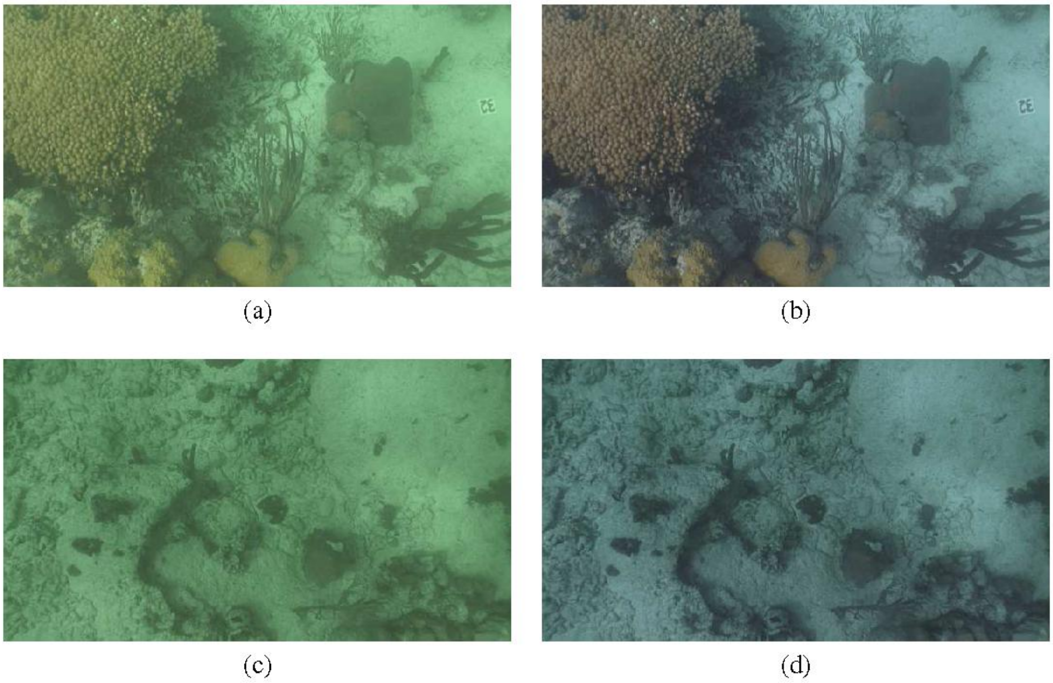


Fig. 3. Sample images from the dataset: (a) and (c) original images depicting different areas of the scene, (b) and (d) same images after contrast enhancement and color correction.

positions originate mainly in small inaccuracies in the 2D feature tracking. This aspect is more accentuated in underwater environments, where scattering and light attenuation decrease image contrast and dim the textures (see Fig. 3). Fig. 7a illustrates the quality (saliency) of the interest points. The saliency is measured by quantifying the amount of the texture information in the neighborhood around the interest points. For instance, sandy regions in the left and right extremities of the scene yield low saliency features, while the coral reef in the central part having richer textures yield higher saliency features (refer to Fig. 8a for a better understanding of the scene configuration).

In other words, higher saliency indicates more discriminative, better localized features. Fig. 7b illustrates the back-projection error of the vertices while Fig. 7c represents the reconstruction error versus the Bundle Adjustment. It both cases it can be noted that the there is a strong relation between the saliency of the tracked 2D features and the precision of the 3D vertices.

Using the set of 3D vertices, we applied the proposed ortho-mosaic technique (see Section II-D). The result is a high resolution (4 MP), high quality 2D map of the scene, shown in Fig. 8a. By generating a dense grid that approximates the scene surface and texture-mapping it using the ortho-mosaic, we obtain an accurate 3D model of the scene (Fig.8b and c). In this case, we used bilinear interpolation for generating the surface, as it provides a more realistic look, taking into account the geometry of the scene.

IV. Conclusions

This paper presented a complete framework for 3D structure recovery and ortho–mosaicing from video sequences. The method was applied on monocular underwater imagery for seafloor modeling. Moreover, the developed techniques can be readily applied on monocular or stereo imaging systems for applications ranging from autonomous robot navigation to aerial and land–based imagery.

The proposed SFM algorithm uses an image-to-scene association approach that allows a direct recovery of camera pose. Experimental results show that this approach reduces errors in both camera pose and structure estimation by taking into account additional information (i.e. trajectory overlaps). In addition, directly recovering the camera pose naturally copes with scene occlusions and pose estimation errors.

Consequently, we show how precision of the 3D model and camera trajectory estimation is directly related to the saliency of the 2D image features.

Ongoing and future work includes the use of visual vocabularies for more efficient association between image observations and 3D scene model. This is expected to reduce the time complexity of the camera pose estimation and increase the effectiveness of loop-closure detection over large loops.

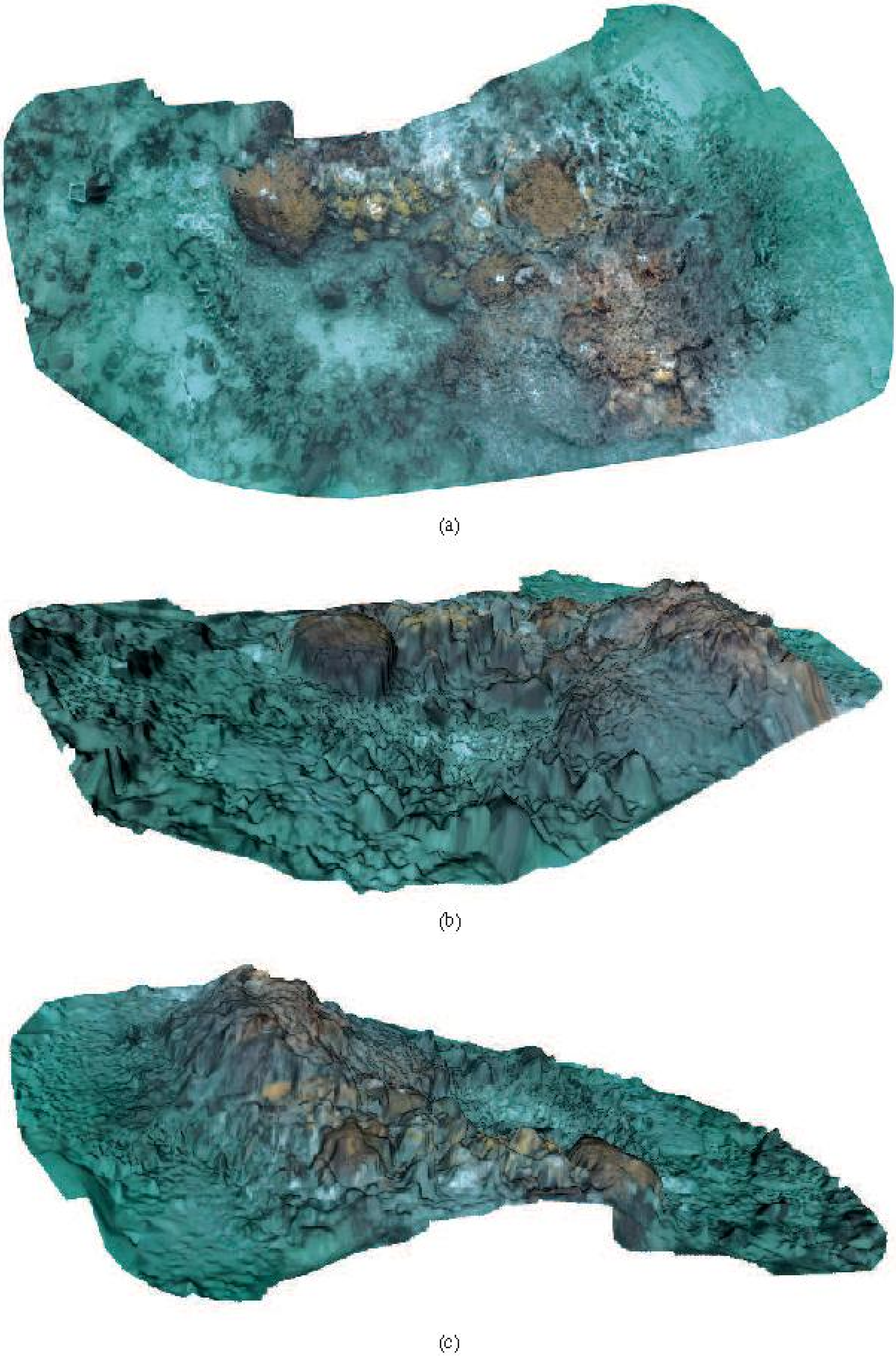
Acknowledgements

This work has been partially funded through the MOMARNET EU project MRTN-CT-2004-505026, and in part by the Spanish Ministry of Education and Science under grant CTM2007-64751. TN has been funded by Generalitat de Catalunya under an FI grant.

The authors would like to thank R. P. Reid and B. Gintert from the University of Miami for providing the Andros dataset (obtained under SERDP Proj CS -1333) .

References

[1] R. Hartley and A. Zisserman, *"Multiple View Geometry in Computer Vision"*. Cambridge University Press, ISBN: 0521540518, 2004.



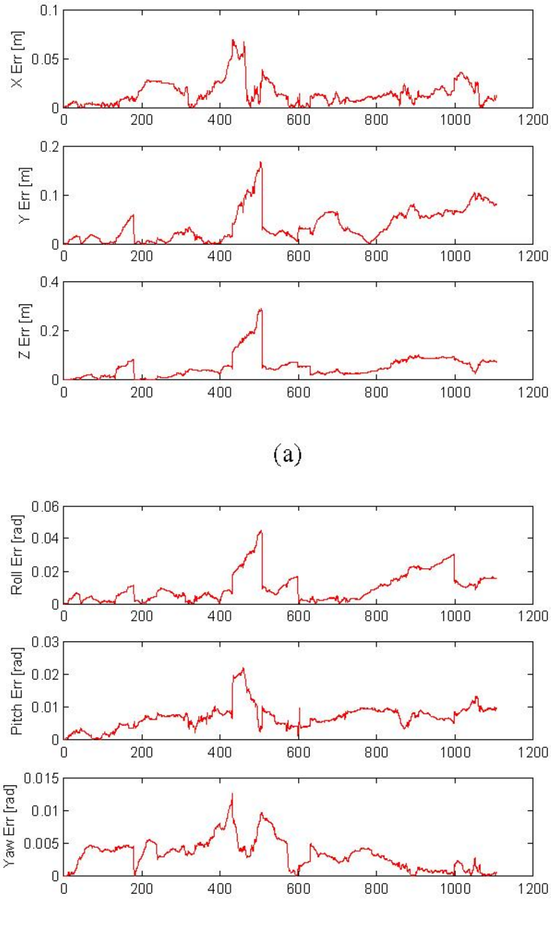
(a)

(b)

(c)

Fig. 8. Obtained scene model: (a) 2D ortho-mosaic of the scene, (b) texture-mapped 3D model of the scene and (c) another view of the 3D model.

978-1-4244-2126-8/08/$25.00 ©2008 IEEE



(a)

(b)

Fig. 6. Camera pose estimation errors vs. BA: (a) position errors in X, Y and Z, (b) attitude error in ϕ, θ and ψ.

[2] N. Gracias, S. Zwaan, A. Bernardino, and J. Santos-Victor, "Mosaic based Navigation for Autonomous Underwater Vehicles," *Journal of Oceanic Engineering*, vol. 28, no. 4, October 2003.

[3] R. Garcia, T. Nicosevici, and X. Cufí, "On the way to solve lighting problems in underwater imaging," in *MTS/IEEE OCEANS Conference*, 2002, pp. 1018–1024.

[4] M. Kudzinava, "”feature-based matching of underwater images”," Master's thesis, May 2007.

[5] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of Alvey Conference*, Manchester, UK, August 1988, pp. 189–192.

[6] P. Beaudet, "Rotationally invariant image operators." in *Proceedings of the 4th ICPR: International Joint Conference on Pattern Recognition*, 1978, pp. 579–583.

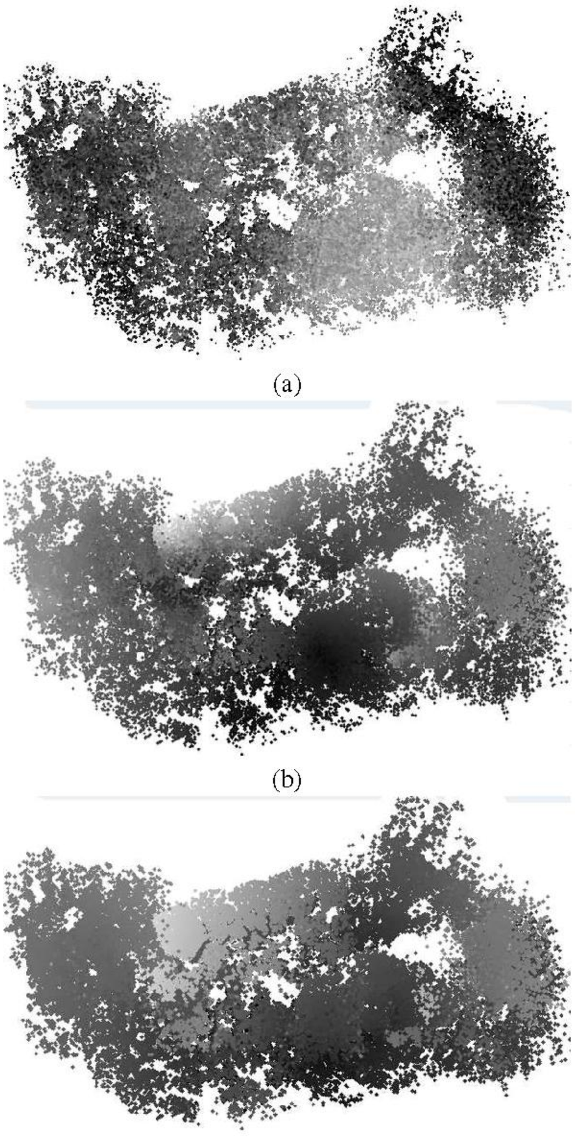
[7] T. Lindeberg, "Feature detection with automatic scale selection," CVAP: Computational Vision and Active Perception Laboratory, Tech. Rep. 30, 1998.

[8] A. Baumberg, "Reliable feature matching across widely separated views," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. 774–781.

[9] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[10] H. Bay, T. Tuytelaars, and L. J. V. Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision*, 2006, pp. 404–417.

[11] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Conference on Computer Vision*, vol. 65.



(a)

(b)

(c)

Fig. 7. Comparison between feature saliency distribution and 3D vertex precision: (a) Feature saliency, lighter valuers correspond to higher quality features, (b) 3D vertex average back-projection error, lighter valuers correspond to higher errors and (c) 3D vertex position error vs. Bundle Adjustment.

[12] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching," *Journal of the ACM*, vol. 45, pp. 891–923, 1998.

[13] T. S. Huang and O. D. Faugeras, "Some properties of the e-matrix in two-view motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1310–1312, 1989.

[14] X. Armangue and J. Salvi, "Overall view regarding fundamental matrix estimation," *Image and Vision Computing*, vol. 21, pp. 205–220, 2003.

[15] M. I. A. Lourakis and R. Deriche, "Camera self-calibration using the singular value decomposition of the fundamental matrix: From point correspondences to 3D measurements, Tech. Rep. RR-3748.

[16] L. Robert and O. D. Faugeras, "Relative 3D positioning and 3D convex hull computation from a weakly calibrated stereo pair," in *International Conference on Computer Vision*, 1993, pp. 540–544.

[17] B. Triggs, "Autocalibration from planar scenes," in *European Conference on Computer Vision*, vol. 1406, 1998, pp. 89–105.

[18] Y. Ma, S. Soatto, J. Kosecka, and S. Shankar Sastry, *"An invitation to 3-D Vision"*. Springer, 2003.