3D Large-Scale Seabed Reconstruction for UUV Simultaneous Localization and Mapping

Yvan R. Petillot*, Joaquim Salvi, Elisabet Batlle****

** Ocean Systems Lab, Heriot-Watt University, EH14-4AS*
Edinburgh, Scotland (Tel: +44 (0)131 451 8277; e-mail: Y.R.Petillot@hw.ac.uk).
*** Computer Vision and Robotics, University of Girona, E-17071*
Girona, Spain (Tel: +34 972 41 8483; e-mail: qsalvi@eia.udg.edu).

Abstract: This paper proposes a new technique to reconstruct large 3D scenes from a sequence of video images by combining Bayesian filtering and state-of-art 3D computer vision. The approach performs the alignment of a sequence of 3D partial reconstructions of the seafloor thanks to the re-observations of passive landmarks by means of a linear Kalman filter-based SLAM approach. Landmarks are detected on the images and characterized considering 2D and 3D features. Landmarks are re-observed while the robot is navigating and data association becomes easier but robust. Preliminary results are performed in virtual scenarios but processing real images synthetized from underwater textures.

Keywords: Visual SLAM, Surface Registration, Stereo Vision, Underwater Imaging, Computer Vision, Simultaneous Localization and Mapping.

1. INTRODUCTION

Optical underwater imaging is a key technology for scientific research and industrial developments since it provides high resolution images of underwater structures. Unfortunately, light is strongly attenuated and scattered in water, limiting the field of view of cameras to a few meters. This has limited the use of optical systems and traditionally, acoustic sensors have been preferred. Nowadays, however, high resolution video sequences can be acquired from short distances using underwater vehicles. Therefore, video cameras are often mounted on underwater vehicles to survey areas of interest at short range. In the process hundreds of images showing partial views of the scene are collected. These images have to be manipulated to deliver a unique large scale map.

Recent underwater imaging research has mainly focused on the alignment of planar views, i.e. mosaicing, while the major areas of interest usually contain 3D structure. Examples of relevant applications are the survey of benthic habitats where aquatic organism live such as sea grass and coral reefs, reconstruction of hydrothermal vent fields such as the one discovered in the Mid-Atlantic Ridge, ancient and modern shipwrecks and archaeological settlements and 3D man-made underwater structures in need of regular inspection or used for docking. Mosaics of 3D structures suffer from misalignment that deteriorates the mapping. Besides, 3D registration techniques (similar to ones applied in reverse engineering) are inapplicable because 3D point's position is noisy, points are sparse and point resolution is not constant. However, Simultaneous Localization and Mapping (SLAM) performs well basically due to its intrinsic capability to deal with uncertainty in vehicle location and mapping. In the Simultaneous Localization and Mapping problem the vehicle starts in an unknown location in an unknown environment and proceeds to incrementally build a navigation map of the environment while simultaneously using this map to update its location. The SLAM community has focused on optimal Bayesian filtering and there exist many methods available, especially in indoor environments using laser scanning (Estrada et al, 2005), sonar (Leonard et al., 2001) and video (Newman et al., 2005). Unsurprisingly, very few papers have tackled SLAM in underwater. The ones that tried, they have always focused on acoustic data (Mahon et al. 2004; Tena-Ruiz et al. 2004).

The key to a successful visual SLAM-based system underwater must lie in the selection of very robust landmarks so that data association is possible even under different view points and illumination patterns. The second important factor to take into account is the likely sparseness of image points, due to the environment and the necessary selection of robust features.

This paper explores a solution to this problem using video stereo images and 3D state-of-art computer vision. The technique filters the navigation data of the vehicle using a stochastic map. The stochastic map keeps the estimates of landmarks whose re-observation is used to aid the localization of the vehicle. The stochastic map is smoother using a Rauch-Tung-Striebel (RTS) smoother. Once the vehicle has completed the journey, local 3D surfaces acquired whilst the robot was moving are aligned delivering a unique and accurate registered surface of the seabed.

The paper first describes the Kalman filter-based SLAM approach we propose. Second, the RTS smoother is detailed. Third, local 3D surface acquisition is explained in section 3. Then, landmark detection and characterization is explained in section 4. The article presents some preliminary results in a virtual scenario in section 5.

1 SIMULTANEOUS LOCALIZATION AND MAPPING

In our framework, landmarks consists of 3D points in (X,Y,Z). The position and velocity of the vehicle together with the position of landmarks are also measured with respect to (X,Y,Z), the system of equations is linear and can be modelled by a linear Kalman Filter. This is a key advantage of our approach. We are therefore proposing to use a classical stochastic map approach (Smith et al., 1990). The stochastic map is augmented to accommodate new landmarks as they are observed. The stochastic map also stores and maintains all the covariances and correlations between states. With fully correlated landmarks, the re-observation of any landmark aids to correct the whole map and filter the trajectory of the vehicle.

A Kalman filter is composed of three steps: Prediction, Observation and Update. We have added a fourth step to incorporate new landmarks to the state of the filter. The four steps are shown in Fig. 1 and explained in the following.

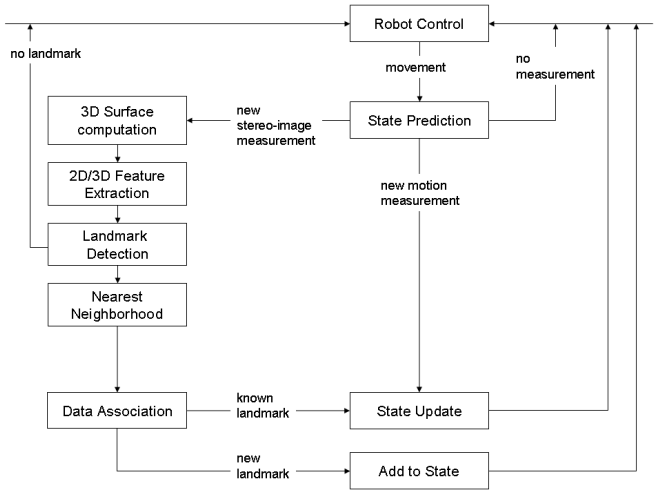

Fig. 1. Flow diagram of the SLAM module.

1.1. Process Model

The state of the system consists initially of the position x and velocity $\dot{x}$ of the vehicle. Once a landmark is observed, the state is augmented with the position x_j of the new landmark. Landmarks are kept during the whole mission of the vehicle. Hence, the state of the system at instant k is defined by the following equation,

$$x(k) = \left[x, \dot{x}, x_1 \ldots x_n\right]. \quad (1)$$

Assuming that the state at instant k is known, the prediction of the next state is modelled by

$$\hat{x}(k+1 \mid k) = F(k)\hat{x}(k \mid k) \quad (2)$$

where $F(k)$ is the state matrix, so that he velocity of the vehicle and landmarks are assumed constant. The position of the vehicle follows a standard linear model. Then, the predicted covariance matrix is

$$P(k+1 \mid k) = F(k)P(k \mid k)F^T(k) + Q(k) \quad (3)$$

where Q is the process noise matrix. It consists of a diagonal of 0 except in the terms of vehicle position and velocity where the corresponding process noise variances are added. Both noise variances are fixed, determined off-line and define the reaction of the filter to sudden changes of the ground truth position/velocity of the vehicle; and the covariance matrix at the initial time stamp $P(1|1)$ is defined by the noise variance of the position, the variance and velocity measuring noise given by the navigation data and the variance of the landmark measurement noise given by the video camera.

1.2. Observation Model

For an underwater vehicle, a classical navigation system includes a Doppler Velocity Log (DVL) which estimates the vehicle speed over ground at high frequency (100Hz). Landmarks are observed by the stereo camera with respect to the current position of the vehicle and bound the navigation errors introduced by the intrinsic drift of sensors when loop closing happens. We consider that all landmarks are stationary and due to our data association process a single landmark is at the most observed at a given instant of time (see section 4). Besides the predicted observation of the vehicle motion is

$$\hat{z}_m(k+1) = H_m\hat{\dot{x}}(k+1 \mid k) \quad (4)$$

where H_m is an identity matrix since we assume that speed is constant; and the predicted observation of landmark j is

$$\hat{z}_j(k+1) = H_j\hat{x}_j(k+1 \mid k) \quad (5)$$

where H_j computes the relative position of landmark j by subtracting the predicted position of the landmark by the predicted position of the vehicle at instant of time $k+1$.

When a new observation of the vehicle motion z_m is delivered by the vehicle or a new landmark z_j is re-observed by the video camera, the innovation vector is computed accordingly in the following way,

$$v(k+1) = \left[z_m(k+1) \;\; z_j(k+1)\right]^T - \left[\hat{z}_m(k+1) \;\; \hat{z}_j(k+1)\right]^T \quad (6)$$

together with an associated innovation covariance matrix given by:

$$S(k+1) = H(k+1)P(k+1 \mid k)H^T(k+1) + R(k+1) \quad (7)$$

where $H(k+1) = [H_m \;\; H_j]$ depends on whether the motion and/or any landmark is observed at $k+1$; and $R(k+1)$ is the measurement noise matrix defined as a diagonal matrix containing the vehicle motion measurement noise variance and the landmark position measurement noise matrix at time $k+1$.

1.3. Process Update

The estimate of the state vector and its corresponding covariance matrix are then updated as follows,

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + W(k+1)v(k+1) \quad (8)$$

and

$$P(k+1|k+1) = P(k+1|k) - W(k+1)H(k+1)P(k+1|k) \quad (9)$$

where

$$W(k+1) = P(k+1|k)H(k+1)S^{-1}(k+1) \quad (10)$$

is known as the optimal Kalman gain at time *k+1*.

1.4. Adding new landmarks

New landmarks are introduced in the filter state just after the process update step, since all vectors and matrices forming the filter have to be updated to use the new landmark in the filtering process.

So, when a new landmark is observed:

- the observed position is added to the state *x(k +1|k + 1)*;
- the covariance matrix *P(k + 1|k + 1)* is enlarged by adding the rows and columns corresponding to the new landmark. The vehicle position variance at that time together with the landmark measurement noise variance is used to initialize the variance of the landmark in the filter;
- the state matrix *F(k+1)* is enlarged by adding *1*'s to the corresponding landmark position;
- the process noise matrix *Q* is enlarged adding *0*'s, since landmarks are stationary;
- the *H(k + 1)* measuring matrix is enlarged so that the relative position of the new landmark can be predicted at the next time step; and
- the matrix measurement noise *R(k + 1)* is also enlarged adding the landmark measurement noise variance accordingly.

2. RAUCH-TUNG-STRIEBEL SMOOTHER

The Kalman filter uses all measurements up to the last iteration to estimate the state at the last iteration. The Rauch-Tung-Striebel (RTS) smoother uses all the measurements before and after each iteration to estimate the state at each iteration. It is a post-processing filter that works on the stored outputs of the Kalman filter by re-processing them. The smoother works by combining a forward pass filter with a backward pass filter. It was originally designed to work with fixed size state vectors. However, the stochastic map adds new states to the state vector as it observes new landmarks. The algorithm adapts the RTS fixed-interval smoother to work with the stochastic map by fixing the size of the state vector to the size of the stochastic map on the last iteration. The output of the RTS has been shown to improve the accuracy of the stochastic map solution as well as providing smoother trajectories (Tena-Ruiz et al. 2004).

So, once the Kalman filter has finished, we fix *k* to the instant of time *n-1* and we go backwards till we reach instant of time *1*. The predicted smoother state is computed in the following way:

$$\hat{\tilde{x}}(k+1|k) = F(k)\hat{x}(k|k) \quad (11)$$

and the predicted covariance matrix as follows

$$\hat{\tilde{P}}(k+1|k) = F(k)P(k|k)F^T(k) + Q \quad (12)$$

Then, the smoother gain matrix *J* is computed as follows

$$J(k) = P(k|k)F^T(k)\hat{\tilde{P}}^{-1}(k+1|k) \quad (13)$$

and, hence, the filtered state is given by the following equations:

$$\tilde{x}(k|k) = \hat{x}(k|k) + J(k)\left(\tilde{x}(k+1|k+1) - \hat{\tilde{x}}(k+1|k)\right) \quad (14)$$

$$\tilde{P}(k|k) = P(k|k) + J(k)\left(\tilde{P}(k+1|k+1) - \hat{\tilde{P}}(k+1|k)\right)J^T(k) \quad (15)$$

We initialize the smoother so that

$$\tilde{x}(n|n) = \hat{x}(n|n) \qquad \tilde{P}(n|n) = P(n|n) \quad (16)$$

3. LOCAL 3D SURFACE ACQUISITION

The problem addressed here is to recover 3D structure from a video stereo pair mounted on an underwater vehicle with changing illumination and an unknown surface structure. We have decided to use a wide-baseline stereo approach as depicted in Fig.2.

First, An Homomorphic filter is used to normalize the brightness across the image and compensate for non uniform lighting patterns. This is followed by a Contrast-Limited Adaptive Histogram Equalization (CLAHE) to enhance the contrast of images. CLAHE operates on small data regions of the image. A further bilinear interpolation is performed to remove artificially induced boundaries between regions. Finally, an Adaptive Noise-Removal Filtering is carried out to remove the noise produced by the equalization especially in those areas with small variance (constant brightness). The resulting images are brighter, better contrasted and normalized. This facilitates the comparison of two images acquired at different times and viewpoints, enabling the matching of image features. Applying this process the number of features detected in the image is multiplied by ten times and features are spread throughout the whole image, which it is quite satisfactory.

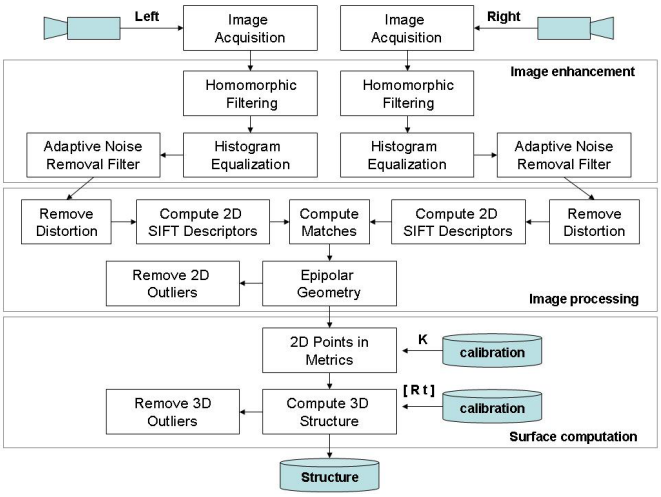

Fig. 1. Flow diagram detailing Image Enhancement, Image Processing and Surface Computation modules and their corresponding tasks to compute surface structure from raw images.

In order to get the metrics from two stereo images, both cameras need to be calibrated obtaining the intrinsic matrices of both cameras K_1 and K_2 and the relative transformation *[R t]* of camera *1* (left) with respect to camera *2* (right). At this point, matrices K_1 and K_2 are used to rectify both images removing lens distortion.

Then, we use the Scale Invariant Feature Transform (SIFT) proposed by Lowe (Lowe, 2004) to extract distinctive image features. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. This is ideal for wide-base line stereo matching.

Once the matches between both images are obtained by SIFT, we compute the Fundamental matrix to remove false matches not detected by SIFT. Note that it is preferable to be strict at this point removing some correct matches instead of allowing false matches to proceed deteriorating the 3D Reconstruction. Since the whole system is calibrated, the Fundamental matrix is computed by:

$$F=K_2^{-T}RTK_1^{-1} \qquad m_2^T F m_1 = 0 \qquad (17)$$

where m_2 and m_1 are the 2D points of the form $(x,y,1)^T$ in pixels, respectively; and T is the skew matrix of the translation vector t. Then, we remove all those matches that do not lay on their corresponding epipolar lines.

Furthermore, we compute the disparity between the remaining 2D points and we remove those whose disparity is larger than 3σ, where σ is the square root of the standard deviation of the disparity distribution. This process permits the removal of remaining outliers, since usually outliers suffer large disparity discrepancies.

Once the set of correct matches has been obtained, the 3D structure can be extracted by using a linear triangulation. So, first we transform the pixels to metric coordinates,

$$\hat{m}_1 = K_1^{-T} m_1 \qquad \hat{m}_2 = K_2^{-T} m_2 \qquad (18)$$

and then, we compute matrix A_i for every pair i of points as follows,

$$A_i = \begin{pmatrix} 0 & -1 & \hat{y}_{1i} & 0 \\ -1 & 0 & \hat{x}_{1i} & 0 \\ (-R_2 + \hat{y}_{2i} R_3) & & -t_y + \hat{y}_{2i} t_z & \\ (-R_1 + \hat{x}_{2i} R_3) & & -t_x + \hat{x}_{2i} t_z & \end{pmatrix} \qquad (19)$$

where $R = (R_1,R_2,R_3)^T$, $t=(t_x,t_y,t_z)^T$, and *[R t]* is the rotation and translation of camera *1* with respect to camera *2*. Finally, we perform Singular Value Decomposition obtaining $A_i=U_iD_iV_i^T$. The 3D point M_i corresponds to the fourth column of V_i before normalization (Ma et al, 2004). M_i is measured with respect to camera *1*.

Finally, we remove isolated 3D points as the ones that have less than 2 neighbours in a certain range distance. Isolated 3D points are not desirable since they introduce large residues in the re-observations of landmarks (see section 4.2). The whole process permits the acquisition of a local 3D surface of the imaged seabed measured with respect to the current vehicle position.

4. DATA REPRESENTATION

Let *X(k)* be the position of the vehicle at time *k* in its six degrees of freedom. Assuming a rigid body motion for the vehicle, the position of the vehicle with respect to a fix reference is a the combination of a rotation *R(k)* and a translation *t(k)*. A partial reconstruction *S(k)* of the surface can be associated to each vehicle position *X(k)*. If a partial reconstruction is not possible at this time (bad visibility, lack of structure in image), a void surface is stored. The 3D large scale *S* can be computed as the union of the partial reconstructions in a global reference frame as: $S = \cup[R(k)\ t(k)]S(k)$.

4.1 Landmark characterization

A landmark is represented by the cloud of 3D points and their corresponding 2D SIFT descriptors in camera *1*. Once the landmark is stored, we also compute landmark position as the gravity centre of the cloud of 3D points. Landmark positions are kept with respect to a world reference (usually the initial position of the vehicle). Note that the Kalman filter computes the position of the vehicle with respect to world reference and that the position of the camera *1* with respect to the vehicle is known by calibration. A partial reconstruction *S(k)* is selected as a landmark only if the number of 2D points is significative and well spread in the image. This criterion avoids the detection of landmarks in poor textured images. Note that the amount of features per landmark is important in data association. Finally, a new landmark can only be detected if it is at a certain distance of already stored landmarks ensuring that at maximum one landmark is detected per image, keeping the algorithm simple but yet reliable.

4.2. Data association

Each time a new partial reconstruction is obtained, we first check if there are any landmarks in the vicinity. Vicinity is determined as a function of the camera field of view (range and aperture); the navigation data uncertainty; and the covariance matrix of the Kalman filter that determines the uncertainty of every landmark position. For every detected landmark, we match the SIFT descriptors of the current 3D local reconstruction to those of the detected landmark obtaining a number of matches. Then, we compute the Fundamental matrix to remove false matches not detected by SIFT. Note that in this case we need to use a Fundamental matrix estimator since although the relative transformation between both images is given by the Kalman filter, it is very imprecisely known to be used in such computation. We have used as F estimator the technique of Least Median of Squares (LMedS) based on Singular Value Decomposition and point data normalization, which has been proved to perform well compared to other *F* estimators (Armangue et al., 2003). Then, we remove false matches and, finally, we keep as a potential re-observation the landmark in the vicinity that maximizes the number of inliers.

For every 2D matches, its corresponding 3D point is known. So, we now have two clouds of 3D points and we can compute the transformation between the two clouds. First, the landmark points are transformed to the vehicle current frame so that now both clouds are in the same reference. Then, the relative transformation *[R t]* between both clouds of points is computed using the method proposed by Mian (Mian et al. 2006). The re-observed landmark position L_C in the current vehicle frame is then $L_C = RL_S + t$, where L_S is the stored landmark gravity centre in the current vehicle frame.

5. EXPERIMENTAL RESULTS

The experiment is so far based on a simulation but includes the entire image processing aspects, though we are dealing with a virtual 3D scenario. The virtual scenario consists in a 3D height map that can be generated by the user or imported from existing data. A texture image can then be wrapped on the 3D virtual surface. Note that texture appearance is deformed according to surface structure. Finally, the ground truth trajectory of the underwater vehicle is generated as a sequence of way points. The simulator interpolates the vehicle trajectory delivering a Navigation table that contains ground truth position and velocity at every time stamp.

Two virtual cameras have been modelled, according to the intrinsic and extrinsic parameters of real cameras, and coupled to the vehicle. At every instant of time, we synthesize both virtual images observed by each camera using ray tracing: the optical ray for every image pixel is computed and intersected with the 3D surface and the intensity value for each pixel is calculated as a linear interpolation of the grey value of the four closest points on the surface texture.

A linear Kalman filter was programmed according to section 1. The vehicle is estimating its position and velocity while is moving in the virtual world and 3D partial reconstruction are acquired in parallel. From time to time, landmarks are re-observed and used to feed the filter so that the trajectory of the vehicle is corrected and, hence, the 3D partial acquisitions of the seabed better aligned. Finally, the RTS smoother is used to smooth the trajectory of the vehicle and obtain an even better alignment.

In the first experiment, the vehicle is describing a trajectory composed of *523* via points. Gaussian noise with zero mean has been added to position ($\sigma_x^2 = 1m^2$) and velocity ($\sigma_{\dot{x}}^2 = 0.025m^2/s^2$). Fig.3 shows how the technique filters the trajectory and it is able to align the *523* partial reconstructions of the seabed. The 3D surface has been interpolated and re-sampled from the *39,522* 3D points obtained by the algorithm.

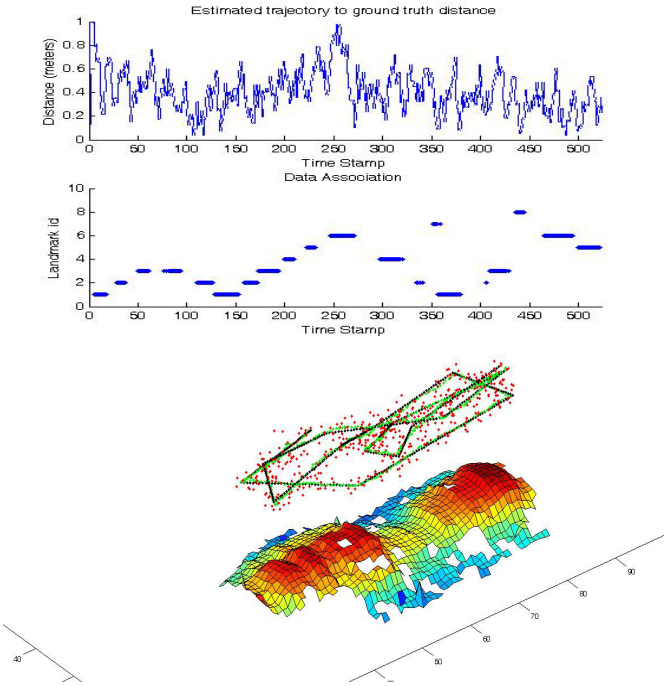

Fig 3. Experiment 1: 3D Reconstruction from 523 via points, noisy vehicle position and velocity. The figure shows the trajectory error versus data association (up) and the reconstructed surface (bottom).

In the second experiment, vehicle position is not measured and we have now added Gaussian noise with a large bias to the measurement of the vehicle velocity ($\mu = 0.05m/s$, $\sigma_{\dot{x}}^2 = 0.001m^2/s^2$). This experiment has been performed to check how the SLAM approach is able to readjust vehicle trajectory thanks to the re-observation of landmarks even in the presence of large bias. Now the vehicle is performing a trajectory composed of *916* via points and detects up to *26* landmarks during the journey. Fig. 4 shows the unfiltered trajectory, the SLAM filtered trajectory and the RTS smoothed trajectory compared to ground truth; and the trajectory error versus the detection of landmarks to see how the error is reduced every time a landmark is re-observed. Fig. 5 shows the reconstruction results obtained with the filtered and smoothed trajectory after the interpolation of a cloud of *53,996* 3D points.

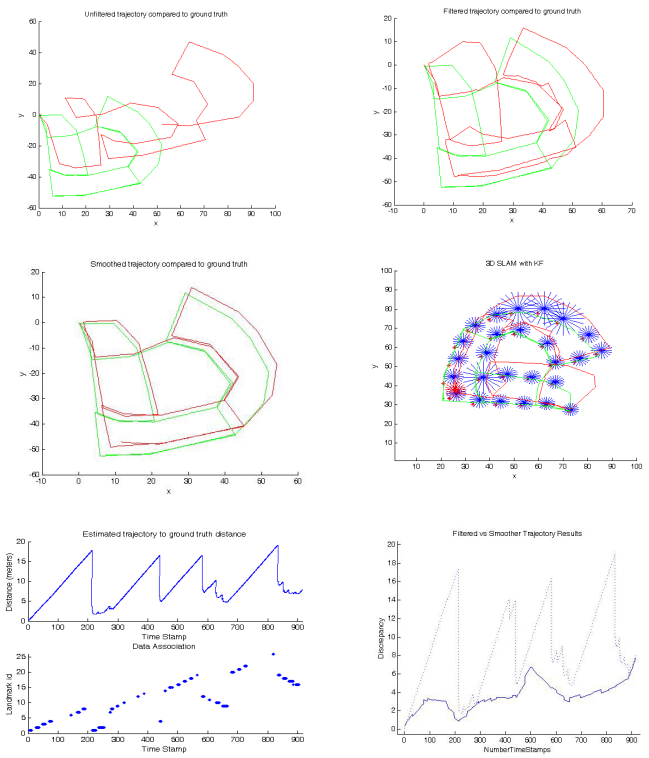

Fig. 4. Experiment 2 (from left to right and top to bottom): a) the unfiltered trajectory (red) to ground truth (green); b) the SLAM filtered trajectory (red) to ground truth (green); c) the RTS smoothed trajectory (red) to ground truth (green); d) the last step of the SLAM filter showing landmark covariances (blue); e) the error in the trajectory compared to ground versus the detection of landmarks; and, finally, f) the RTS smoothed trajectory (solid line) compared to the SLAM filtered trajectory (dotted line).

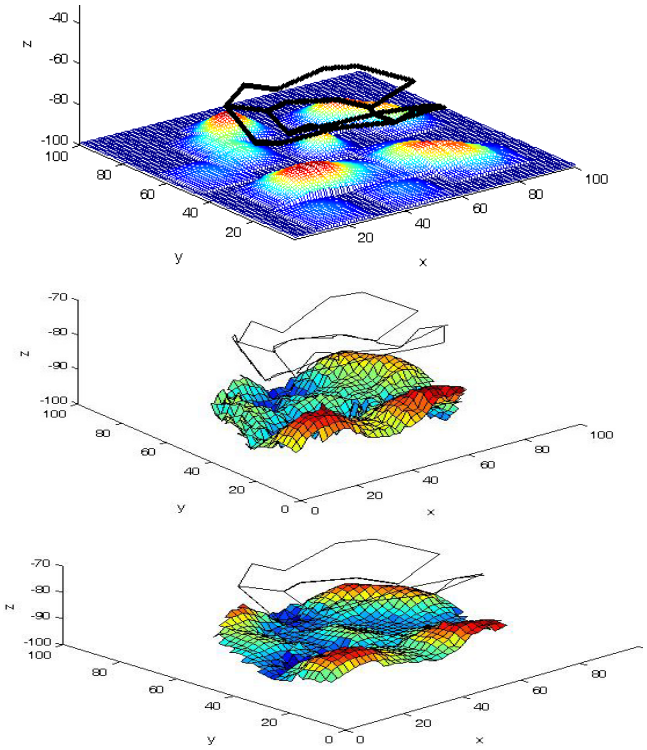

Fig. 5. Experiment 2. 3D Reconstruction results: a) The ground truth; b) The 3D seabed reconstruction by means of the SLAM filtered trajectory; c) The 3D seabed reconstruction by means of the RTS smoothed trajectory.

6. CONCLUSIONS

This paper has presented an approach to perform the 3D reconstruction of the seabed from the alignment of hundreds of partial reconstructions thanks to Simultaneous Localization and Mapping based on Kalman filtering and benefiting from the navigation data of the underwater vehicle and the re-observation of landmarks by using a unique stereo camera. Experimental results are in an early stage but yet show that SLAM performs well to simultaneously localize and map in 3-Dimensions the seabed correcting the intrinsic drift in vehicle navigation every time a landmark is re-observed. Besides, RTS smoothing is convenient as a post-processing step to filter backwards the trajectory obtained by the Kalman filter obtaining a better estimation of the vehicle trajectory and consequently an even better alignment of the seabed. To the best of our knowledge, this paper is the first that proposes SLAM + RTS to deal with the 3D reconstruction of the seabed by just using video cameras. At the time of submission, we are processing real images to reconstruct the floor of Loch Linnhe (Scotland).

REFERENCES

Armangue X., J. Salvi. Overall view regarding fundamental matrix estimation. Image and Vision Computing 21:205–220, 2003.

Estrada C., J. Neira, J.D. Tardos. Hierarchical SLAM: Real-Time Accurate Mapping of Large Environments. IEEE Transactions on Robotics 21(4):588–596, 2005.

Leonard J.J, P.M. Newman, R.J. Rikoski, J. Neira, J.D. Tardos. Towards robust data association and feature modeling for concurrent mapping and localization. Tenth International Symposium on Robotics Research, 2001.

Lowe D.G. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 2(60):91–110, 2004.

Ma Y., S. Soatto, J. Kosecka, S. Sastry. An Invitation to 3-D Vision: From Images to Geometric Models. Springer-Verlag, 2003.

Mahon I., S. Williams. SLAM using Natural Features in an Underwater Environment. IEEE Int. Conf. on Control, Automation, Robotics and Vision, 2076–2081, 2004.

Mian A.S., M. Bennamoun, R.A. Owens. A Novel Representation and Feature Matching Algorithm for Automatic Pairwise Registration of Range Images. Int. Journal of Computer Vision 66(1):19–40, 2006.

Newman P., K. Ho. SLAM-Loop Closing with Visually Salient Features. IEEE International Conference on Robotics and Automation, pages 635–642, 2005.

Smith R., M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. in Autonomous Robot Vehicles., I. Cox and G. Wilfong, Eds. New York: Springer-Verlag, 1990.

Tena-Ruiz I., S. Raucourt, Y. Petillot, D.M. Lane. Concurrent Mapping and Localization Using Sidescan Sonar. IEEE Journal of Oceanic Engineering 29(2):442–456, 2004.