Towards Robust Image Registration for Underwater Visual SLAM

Antoni Burguera, Francisco Bonin-Font and Gabriel Oliver
Dept. Matemàtiques i Informàtica, Universitat de les Illes Balears, Ctra. Valldemossa Km. 7.5, Illes Balears, Spain

Keywords: Underwater Robotics, Visual SLAM, Data Association, Image Registration.

Abstract: This paper proposes a simple and practical approach to perform underwater visual SLAM. The proposal improves the traditional EKF-SLAM by adopting a Trajectory-based schema that reduces the computational requirements. Linearization errors are also reduced by means of an IEKF. One of the most important parts of the proposed SLAM approach is robust image registration, which is used in the data association step making it possible to close loops reliably. Thanks to that, as shown in the experiments, the presented approach provides accurate pose estimates using both a simulated robot and a real one.

1 INTRODUCTION

A crucial issue in underwater robotics nowadays is the one of *localization*, which consists in determining and keeping track of the robot location in the environment. The so called *Simultaneous Localization And Mapping* (SLAM) (Durrant-Whyte and Bailey, 2006) constitutes the most common and successful approach to perform localization.

Acoustic sensors have interesting properties under the water, such as large sensing ranges, and that is why they are a common choice to perform underwater SLAM (Ribas et al., 2007). Nevertheless, acoustic sensors have lower spatial and temporal resolution than cameras. Thus, cameras are convenient for surveying or intervention applications where the robot has either to navigate close to the bottom or to stay near an object of interest. Examples of such applications are mosaicking or object manipulation (Prats and Ribas, 2012). Moreover, recent literature shows that cameras are used more and more to perform visual SLAM under the water (Eustice et al., 2008).

Accordingly, this study proposes a vision based approach to perform underwater SLAM. More precisely, the proposal in this paper is to integrate information coming from a single, bottom-looking, monocular camera, an altimeter and a dead reckoning sensor by means of SLAM. Thanks to that, accurate estimates of an underwater robot pose will be obtained.

The main advantages of our proposal are summarized next. First, we pay special attention to image registration in order to determine robustly if the robot is returning to previously visited areas (i.e. loop closure). Detecting these situations is extremely important as they provide valuable information to the SLAM process. Second, our proposal is not constrained to constant altitude missions since it uses external altitude information. In this way, the proposed image registration method is able to deal with translation, rotation and scale changes. Third, our approach to SLAM adopts a *Trajectory Based* schema (Burguera et al., 2010), similar to *Delayed State Filtering* (Eustice et al., 2008), in order to reduce the computational complexity. Finally, an *Iterated Extended Kalman Filter* (IEKF) (Bar-Shalom et al., 2001) is used to reduce the linearization errors inherent to standard *Extended Kalman Filters* (EKF).

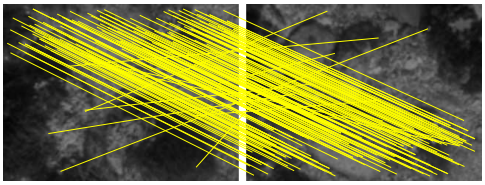
2 IMAGE REGISTRATION

In SLAM, data association refers to the registration of current sensory input to previously gathered data. Successfully registering such pieces of information makes it possible not only to estimate incrementally the robot pose, but also to perform loop closures.

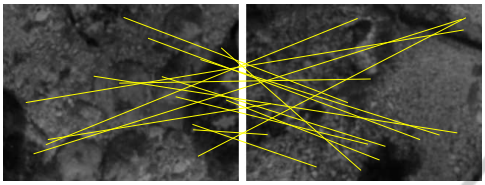
When using vision sensors, data association is tightly related to image registration and usually relies on the detection and matching of image features. Given two images, our proposal to data association starts by searching their features and descriptors according to *Scale Invariant Feature Transform* (SIFT) (Lowe, 2004), although other feature detectors and matchers could also be used.

Feature coordinates, which are found in pixels, are

Burguera A., Bonin-Font F. and Oliver G..
Towards Robust Image Registration for Underwater Visual SLAM.
DOI: 10.5220/0004682005390544
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 539-544
ISBN: 978-989-758-009-3



(a)



(b)

Figure 1: Feature matching using underwater images. Yellow lines represent correspondences between features. (a) Overlapping images (b) Non overlapping images.

then converted to meters by assuming a locally flat floor and providing the distance to the bottom and the camera focals are known. The former is measured by the altitude sensor and the latter can be obtained from the data sheet or through a calibration process. Thanks to this, altitude changes, which are responsible for scale changes between images, are properly taken into account.

Next step is to compute SIFT matchings between the two images. In spite of the robustness of SIFT, the reduced contrast in underwater imagery due to bad illumination conditions, and the fact that most of the gathered images look similar can lead to wrong SIFT matchings. These wrong matchings will influence the motion estimate even if most of the matchings are correct. Figure 1-a exemplifies this very common situation. Also, SIFT, as well as many other feature matchers, are likely to detect matchings even between images corresponding to non overlapping areas, as illustrated in Figure 1-b. These problems have to be solved because wrong image associations, especially wrong loop closings, may lead EKF-SLAM to unrecoverable errors.

Accordingly, a key aspect of our data association method is to determine whether two images overlap or not and, if they do, compute the roto-translation that better explain the correct matchings between them. Our proposal is based on the following premise: correct matchings tend to propose a single roto-translation whilst incorrect matchings do not and thus can be considered outliers. The goal of *Random Sample Consensus* (RANSAC) (Fischler and Bolles, 1981) is, precisely, to find a single model where inliers fit while discarding outliers and that is why RANSAC has been adopted in this study.

Figure 2 shows the proposed algorithm to compute the roto-translation between two underwater images using RANSAC. The symbol $\oplus$ denotes the compounding operator, as described in (Smith et al., 1987). Roughly speaking, this algorithm randomly selects a subset C of the SIFT matchings M and then computes the roto-translation $X = [x, y, \theta]^T$ that better explains them. Next, each of the non selected matchings is tested to check if it fits X with an acceptable error level. If so, it is selected too. Finally, if the number of selected matchings $|C|$ exceeds a certain threshold, the roto-translation that better explains all the selected matchings is computed. After a fixed number of iterations, the best of the computed roto-translations constitutes the output of the algorithm. However, if not enough matchings have been selected in any of the iterations, the algorithm assumes that the two images do not overlap.

Input:
F_{ref}: Features $\{p_1, p_2, \cdots, p_m\}$ in the first image
F_{cur}: Features $\{q_1, q_2, \cdots, q_n\}$ in the second image
M: Matchings $M = \{(i,j) | visual\_matching(p_i, q_j)\}$
$nIter$: Number of iterations to perform
N: Number of matchings to be randomly selected
α: Maximum allowable error per matching
β: Min. number of selected matches to consider a model

Output:
X_{best}: The estimated roto-translation
ε_{best}: The error of the estimated roto-translation
$found$: Boolean stating if reliable matching found

Algorithm:
$k \leftarrow 0$; $\varepsilon_{best} \leftarrow \infty$; $found \leftarrow false$;
while $k < nIter$ **do**
  $C \leftarrow$ random selection of N items from M;
  $(X, \varepsilon) \leftarrow$ find_motion(F_{ref}, F_{cur}, C);
  foreach $(i,j) \in (M - C)$ **do**
    if $\|p_i - X \oplus q_j\| < \alpha$ **then**
      $C \leftarrow C \cup \{(i,j)\}$;
    end
  end
  if $|C| > \beta$ **then**
    $(X, \varepsilon) \leftarrow$ find_motion(F_{ref}, F_{cur}, C);
    if $\varepsilon < \varepsilon_{best}$ **then**
      $\varepsilon_{best} \leftarrow \varepsilon$; $X_{best} \leftarrow X$; $found \leftarrow true$;
    end
  end
  $k \leftarrow k + 1$;
end

Figure 2: RANSAC underwater image registration.

The algorithm relies on the so called *find_motion* function, which takes a set of feature matchings C and feature coordinates in the first (F_{ref}) and second images (F_{cur}) as inputs. This function provides the roto-translation X that better explains the overlap between the images by searching the roto-translation that minimizes the sum of squared distances between the matchings in C. More specifically, the roto-translation X and the associated error ε are computed

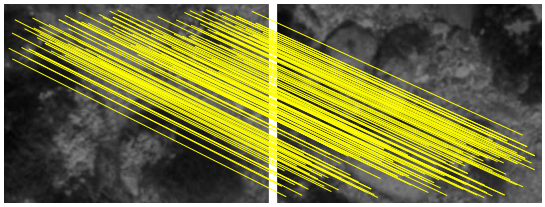


Figure 3: RANSAC underwater image registration.

as follows:

$$X = \underset{x}{\operatorname{argmin}} f(x) \quad (1)$$

$$\varepsilon = f(X) \quad (2)$$

being

$$f(x) = \sum_{\forall(i,j)\in C} ||p_i - x \oplus q_j||^2 \quad (3)$$

where p_i and q_j are feature coordinates in F_{ref} and F_{cur} respectively.

As an example, Figure 3 shows the feature correspondences after applying our proposal to the images previously shown in Figure 1-a. It can be seen how the wrong correspondences have been rejected and only those explaining the true motion remain. Our proposal has also been applied to the images in Figure 1-b and it correctly detected that they do not overlap.

3 VISUAL SLAM

Being based on EKF-SLAM, our approach performs three main steps: prediction, state augmentation and update. During the prediction, the robot pose is estimated by means of dead reckoning. The state augmentation is in charge of storing the newly acquired information. Finally, the measurement step updates the prediction by associating the current image to previously stored data using the described data association algorithm. Our proposal is to perform the measurement update using only one every N frames and thus reducing the computational cost. Henceforth, the used frame will be called a *keyframe* and N will be referred to as the *keyframe separation*.

In this study, similarly to the Trajectory-Based schema, the state vector X_k is defined as follows:

$$X_k = [x_1^0, x_2^1, x_3^2, \cdots, x_k^{k-1}]^T \quad (4)$$

where each $x_i^{i-1} (2 \leq i \leq k)$ denotes a roto-translation from keyframe F_{i-1} to keyframe F_i and x_1^0 represents the initial robot pose relative to a world fixed coordinate frame. Let us assume, without loss of generality, that $x_1^0 = [0,0,0]^T$. Thus, contrarily to other EKF Visual SLAM methods where the visual features themselves are stored in the state vector, our proposal requires much less computational resources by storing only the motion estimates between keyframes.

The pose of the most recent keyframe with respect to the world fixed coordinate frame can be computed as $x_k^0 = x_1^0 \oplus x_2^1 \oplus x_3^2 \oplus \cdots \oplus x_k^{k-1}$. Also, the current robot pose can be computed by composing the last keyframe pose estimate and the dead reckoning information.

3.1 Prediction and State Augmentation

Under the assumption of static environment, the state vector does not change during the EKF prediction step. However, it has to be augmented as follows when a new keyframe is available.

$$X_k^- = [X_{k-1}^+, x_k^{k-1}]^T \quad (5)$$

At this point, x_k^{k-1} is the rough motion estimate provided by the dead reckoning sensors. Also, keyframes are stored outside the state vector.

3.2 The Update Step

Every time a new keyframe is gathered, it is compared with all the previously gathered ones that are within a certain distance using the data association proposed in Section 2. The data association tells whether the new keyframe matches each of the previously gathered ones and, if so, it provides an estimate of the roto-translation between them. This information is used to build our measurement vector Z_k:

$$Z_k = [(z_k^{C1})^T, (z_k^{C2})^T, \cdots, (z_k^{Cn})^T]^T \quad (6)$$

where $C1, C2, \cdots, Cn$ denote the keyframes that actually match the current one and z_k^{Ci} represents the motion estimated by our RANSAC based approach from the keyframe C_i to the most recent one.

In EKF-SLAM, the observation function h_i is in charge of telling how z_k^{Ci} is expected to be according to the state vector X_k^-. Because of the state vector format, this can be computed as follows:

$$h_i(X_k^-) = x_{Ci+1}^{Ci} \oplus x_{Ci+2}^{Ci+1} \oplus \ldots \oplus x_k^{k-1} \quad (7)$$

Figure 4 illustrates the idea of a measurement z_k^{Ci} and the associated observation function h_i. The full observation function is built as follows:

$$h(X_k^-) = [(h_1)^T, (h_2)^T, \ldots, (h_n)^T]^T \quad (8)$$

That is, each item in the full observation function tells how the measurement in the same position in

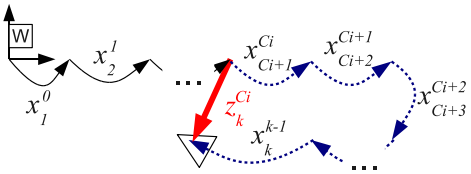


Figure 4: Illustration of a measurement (thick red arrow) and the corresponding observation function (dashed blue arrows)

Z_k was expected according to the state vector. We would like to emphasize that, for each couple of registered images, the whole trajectory portion that connects them is explicitly corrected, contrarily to traditional methods that only explicitly correct the endpoints. For example, all the robot motions depicted as dashed blue arrows in Figure 4 will be corrected by the single measurement z^{Ci}_k.

At this point, the standard EKF update equations, which basically depend on the observation function and the measurement vector, could be used. However, in order to reduce the linearization errors our proposal is to use an IEKF (Burguera et al., 2010). Roughly speaking, the IEKF consists in iterating an EKF and relinearizing the system at each iteration until convergence is achieved. When this happens, the last estimate constitutes the updated state vector X^+_k.

4 EXPERIMENTAL RESULTS

4.1 Simulated Experiments

In order to show the validity of our proposal, we have performed experiments using both a simulated and a real underwater robot. Both the simulated and the real robot software runs on ROS (Quigley et al., 2009), which makes it easy to test software under simulation before deploying it on the real robot and also to analyze the data gathered by the real robot.

For the simulated experiments the underwater robot simulator UWSim (UWSim, 2013) was used. The environment where the simulated robot was deployed was a mosaic of a real sub-sea environment. The pictures shown in Figure 1 are examples of the imagery gathered by the simulated underwater camera.

The simulated mission consisted in a sweeping task, which is very common in underwater robotics. During the mission execution, images coming from a monocular bottom looking camera were gathered as well as the real robot pose, which was solely used as ground truth. Altitude was constant in this simulation. Dead reckoning used in the prediction step was

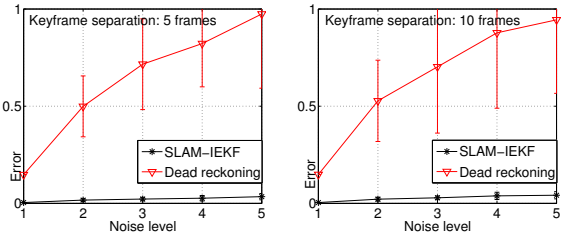


Figure 5: Errors in meters and 2σ bound. (a) Using keyframe separation of 5. (b) Using keyframe separation of 10.

computed by means of visual odometry.

Two different keyframe separations, 5 and 10, have been tested. In our particular test case, a separation of 5 means that, in the straight parts of the trajectory, the overlap between consecutive keyframes is close to 55% of the image. A separation of 10 frames leads to an overlap close to a 10%.

Also, in order to test the robustness of our approach in front of odometric noise, we have added synthetic noise to odometry estimates. Five noise levels have been tested for each keyframe separation. The noise used is additive zero mean Gaussian and the covariance ranges from a $[\Sigma_x, \Sigma_y, \Sigma_\theta] = [0,0,0]$ (noise level 1) to $[\Sigma_x, \Sigma_y, \Sigma_\theta] = [4 \cdot 10^{-5}, 4 \cdot 10^{-5}, 5 \cdot 10^{-4}]$ (noise level 5). The random noise was added to each visual odometry estimate. For each configuration (5 or 10 frames of separation between keyframes) and noise level, 100 trials have been performed in order to obtain statistically significative results. The resulting SLAM trajectories have been compared to the ground truth in order to quantitatively measure their error. The error of a SLAM trajectory is computed as the mean distance between each of the SLAM estimates and the corresponding ground truth pose.

The obtained results are shown in Figure 5. It can be observed that the SLAM error is significantly below the one of dead reckoning. It is clear that the differences due to the keyframe separation and the noise level are very small. Thus, our proposal leads to pose estimates whose quality is almost independent of the dead reckoning noise and the keyframe separation, as long as enough overlapping images are gathered. Also, it is remarkable that the error covariances, which are shown as 2σ bounds in Figure 5, are small and significantly lower than those of dead reckoning. That is, even if very different dead reckoning trajectories are used, the SLAM results are very close to the ground truth.

Figure 6-a shows an example of the results obtained with noise level 2 and a keyframe separation of 10. The figure shows the resulting SLAM trajec-

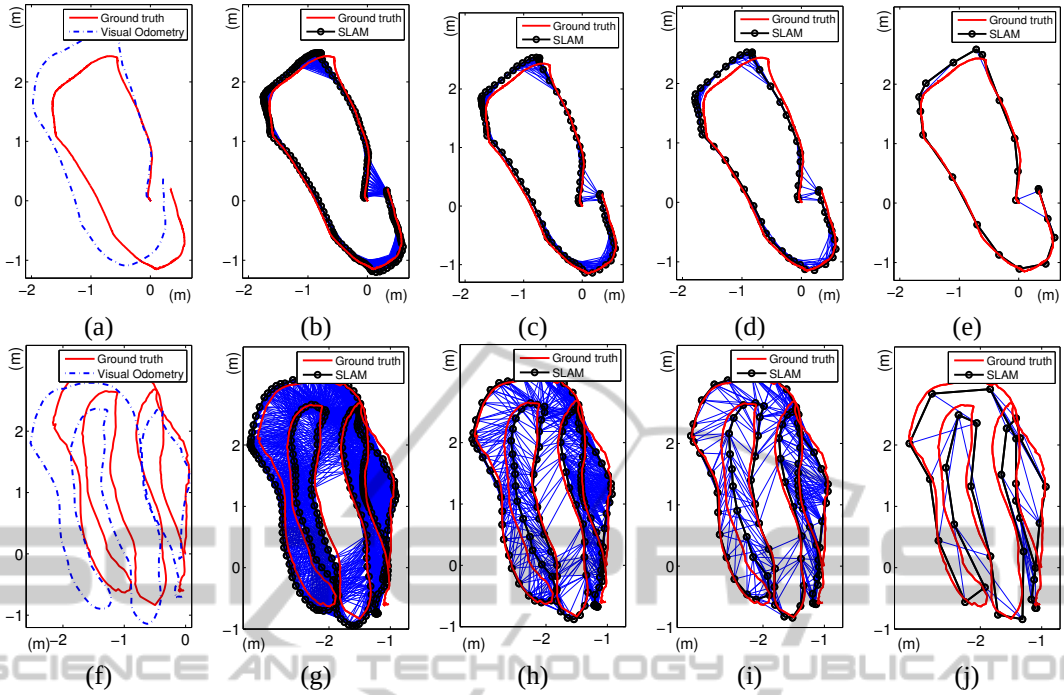


Figure 8: Experiments with the Fugu-C robot. Ground truth and visual odometry are shown for mission 1 (a) and two (f). The resulting SLAM trajectories using keyframe separations of 10, 20, 30 and 90 in mission 1 are shown in (b), (c), (d) and (e). The same results for mission two are depicted in (g), (h), (i) and (j).

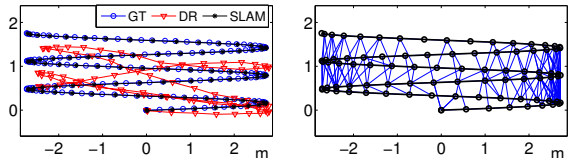


Figure 6: Example of the results obtained with noise level 2 and keyframe separation 10. GT and DR denote Ground Truth and Dead Reckoning. (a) Trajectories (b) Registered images.

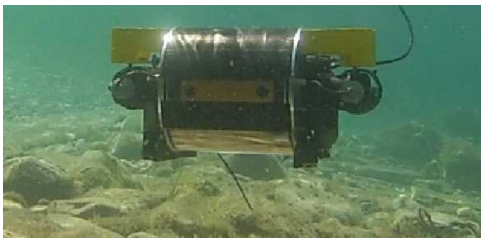


Figure 7: The Fugu-C.

tory, which is almost identical to the ground truth. This is especially remarkable taking into account that the starting dead reckoning data, as it can be seen, is strongly disturbed by noise. Figure 6-b depicts the data associations that have been performed during the SLAM operation.

4.2 Fugu-C Experiments

As for the real robot experiments, they were conducted using Fugu-C (Figure 7), which is a low-cost mini *Autonomous Underwater Vehicle* (AUV) developed by the *Systems, Robotics and Vision* group in our university. The vehicle is equipped, among others, with a nano *Inertial Measurement Unit* (IMU), a pressure sensor and two stereo cameras, one looking forward for obstacle detection and another one looking downwards for computing visual odometry by means of LibViso2 (Geiger et al., 2011). The left camera of the bottom looking stereo pair also provided the imagery to feed our SLAM approach at a rate of 10 frames per second. Moreover, the pressure sensor was used to correct the drift in the altitude estimates provided by the visual odometer.

The experiments were carried out in a pool 7 meters long and 4 meters wide whose bottom was covered with a printed digital image of a real sea bottom. The gathered images were registered to the whole printed digital image to obtain a ground truth for each experiment. Two missions were executed, consisting of a single loop and a sweeping trajectory. The resulting ground truth and visual odometry for these missions are shown in Figure 8-a and 8-f. It can be observed that in both cases, the odometric error is sig-

nificant.

For each mission, different experiments were performed using different keyframe separations. Figures 8-b to 8-e show the resulting trajectories for the first mission using keyframe separations of 10, 20, 30 and 90, which means registering images every 1, 2, 3 and 9 seconds respectively. The results for the second mission under the same conditions are shown in Figures 8-g to 8-j. The lines joining keyframes denote the data associations provided by our RANSAC based approach. Consecutive images have been always registered, although these links have not been depicted for clarity purposes.

In all cases, the resulting trajectories are similar to the ground truth and an important error correction is achieved. The main effect of different keyframe separations is the one of the temporal resolution of the resulting SLAM trajectory but, as long as some images could be registered and loops closed, the pose estimates are close to the ground truth.

5 CONCLUSIONS AND FUTURE WORK

This paper proposes a simple and practical approach to perform underwater visual SLAM, which improves the traditional EKF-SLAM by reducing both the computational requirements and the linearization errors. Moreover, the focus of this paper is the image registration, which is used in the SLAM data association step making it possible to robustly close loops. Thanks to that, as shown in the experiments, the presented approach provides accurate pose estimates both using a simulated robot and a real one.

Nonetheless, the presented approach makes two assumptions that limit the environments where the robot can be deployed. On the one hand, it is assumed that the camera is always poiting downwards. Although the experiments with the real robot show that small changes in roll and pitch are acceptable, avoiding this requirement is one of our future research lines. The simplest way to solve this problem is to use the roll and pitch provided by the gyroscopes in the IMU and use this information to reproject the feature coordinates. On the other hand, our proposal assumes a locally flat floor. Some recent experiments not included in this paper show that our proposal tolerates real oceanic floors that are approximately flat. However, we are now working on solving this issue and fully removing this limitation by using stereo information.

ACKNOWLEDGEMENTS

This work is partially supported by the Spanish Ministry of Research and Innovation DPI2011-27977-C03-03 (TRITON Project), Govern Balear (Ref. 71/211), PTA2011-05077 and FEDER Funds.

REFERENCES

Bar-Shalom, Y., Rong Li, X., and Kirubarajan, T. (2001). *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley and Sons, Inc.

Burguera, A., Oliver, G., and González, Y. (2010). Scan-based slam with trajectory correction in underwater environment. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10)*, Taipei (Taiwan).

Durrant-Whyte, H. and Bailey, T. (2006). Simultaneous localization and mapping (SLAM): part I. *IEEE Robotics and Automation Magazine*, 13(2):99–110.

Eustice, R., Pizarro, O., and Singh, H. (2008). Visually augmented navigation for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*, 33(2):103–122.

Fischler, M. and Bolles, R. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Geiger, A., Ziegler, J., and Stiller, C. (2011). Stereoscan: Dense 3d reconstruction in real-time. In *IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germany.

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

Prats, M. and Ribas, D., e. a. (2012). Reconfigurable AUV for intervention missions: A case study on underwater object recovery. *Journal of Intelligent Service Robotics, Sp. Issue on Marine Robotic Systems*, 5:19–31.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. (2009). ROS: an open source robot operating system. In *ICRA Workshop on Open Source Software*.

Ribas, D., Ridao, P., Tardos, J., and Neira, J. (2007). Underwater slam in a marina environment. *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1455–1460.

Smith, R., Cheeseman, P., and Self, M. (1987). A stochastic map for uncertain spatial relationships. In *Proceedings of International Symposium on Robotic Research, MIT Press*, pages 467–474.

UWSim (2013). UWSim: The underwater simulator. Web. Accessed: 20-June-2013.