

Archaeology via Underwater Robots: Mapping and Localization within Maltese Cistern Systems

Christopher M. Clark
Computer Science Dept.
Cal. Polytechnic State U.
San Luis Obispo, USA
cmclark@calpoly.edu

Christopher S. Olstad
Marine Resources
Development Foundation
Key Largo, FL, USA
chris@mrdf.org

Keith Buhagiar
Dept. of Classics &
Archaeology
University of Malta
keithbuhagiar@onvol.net

Timmy Gambin
AURORA Special
Purpose Trust
Malta
tgambin@hotmail.com

Abstract—This paper documents the application of several underwater robot mapping and localization techniques used during an archaeological expedition. The goal of this project was to explore and map ancient cisterns located on the islands of Malta and Gozo. The cisterns of interest acted as water storage systems for fortresses, private homes, and churches. They often consisted of several connected chambers, still containing water. A sonar-equipped Remotely Operated Vehicle (ROV) was deployed into these cisterns to obtain both video footage and sonar range measurements. Four different mapping and localization techniques were employed including 1) Sonar image mosaics using stationary sonar scans, and 2) Simultaneous Localization and Mapping (SLAM) while the vehicle was in motion, 3) SLAM using stationary sonar scans, and 4) Localization using previously created maps. Two dimensional maps of 6 different cisterns were successfully constructed. It is estimated that the cisterns were built as far back as 300 B.C.

Keywords—ROV, sonar, mapping, SLAM, mosaic.

I. INTRODUCTION

This project concerns the development of an underwater robot system capable of mapping out and navigating underwater tunnel systems. The target environments for this project are cistern networks found in the lower chambers of fortresses and churches in Malta. Archaeologists looking to study and document such systems have found it too expensive and difficult to use people. Furthermore, the human exploration of these subterranean water storage systems is limited by safety and physical constraints and could possibly result in irreversibly damaging to the site under study.

A small underwater robot, or specifically a VideoRay micro-ROV [Remotely Operated Vehicle] was used, (see Fig. 1). Investigators lowered the ROV down well access points until it was submerged in the cistern. The investigators then tele-operated the robot to navigate the tunnels. Two dimensional maps of the cisterns were created using a SeaSprite scanning sonar mounted on top of the ROV. These sonar measurements were used in four ways to develop cistern maps and conduct localization.

The paper is presented as follows. Section II presents related robot mapping techniques. Section III explains the methodology used during the expedition. In section IV, details are provided of the specific mapping and localization techniques. Results from these experiments are shown in section V, followed by conclusions in section VI.



(a)



(b)

Figure 1. Depicted in (a) is the VideoRay Pro III Micro ROV with a SeaSprite sonar mounted on top and the skid removed. In (b), a typical cistern access point is shown.

II. BACKGROUND

Several methods exist for mapping underwater environments when using underwater robots. The maps constructed are used both for the application at hand (e.g. oceanography, marine biology, archaeology, etc.) and to improve the navigation capabilities of the robot itself.

When the robot is localized with respect to some inertial coordinate frame (i.e. the robot's position is known), mapping while in motion is a much simpler task. An approach typically used when operating wheeled robots within indoor environments is to use an occupancy grid map that is updated via the log likelihood approach that assigns a probability of occupation for that each cell in the grid [1].

A common method used for mapping underwater seafloors involves mosaicing bottom images obtained from different locations. Once combined, the resulting mosaic can be used as a map with which the robot can localize itself. Such mapping systems do not rely on the deployment of infrastructure like acoustic positioning systems and do not suffer from drift like IMU based systems. For example, in [2] an ROV was equipped with a real-time mosaicking system. Also, in [3], video mosaicing is used for Autonomous Underwater Vehicle (AUV) navigation.

In recent years, a large amount of research has been conducted in the area of Simultaneous Localization and Mapping (SLAM). SLAM techniques have been developed

and modified for a large number of applications and environments. A good survey of the core techniques including both Kalman Filtering and Particle Filtering based techniques can be found in [4].

One example of robots conducting SLAM in tunnel systems is found in [5]. In that work, the mapping of underground mines was conducted using an autonomous wheeled robot called “Groundhog”.

Other relevant work includes the work conducted in underwater robot SLAM. One of the first instances includes the work done in [6], where sonar scans were used to map and track features of the environment. More recently, successful 3D tunnel mapping in underwater environments was demonstrated in [7].

Unlike the work in [7], this paper describes applications which only permit the passage of small-scale robot systems (i.e. passage opening diameters on the order of 0.3m). Furthermore, the ROV was equipped only with a depth sensor, compass and scanning sonar. To overcome this limitation in sensing, a dynamic model of the ROV was used for the prediction step of both the SLAM and Particle Filter localization algorithm.

A major issue associated with this approach is that tether snags and collisions with walls are not considered in a typical dynamic model. Such occurrences are accounted for and it is shown that when implemented within a particle filter based SLAM approach, i.e. FastSLAM [4], mapping is possible even when frequent tether snags or collisions occur.

III. EXPERIMENT DESCRIPTION

Eight different sites in Malta and Gozo were visited. The cisterns were estimated to be constructed between 300 BC and the 15th or 16th century. At each site, the ROV was initially lowered through a small opening and then down a 3-15 meter deep chute before submerging in the cistern. As shown in Fig. 2(a), several layers of construction can be observed with increasing depth. A reflection of the ROV’s two lights can be seen on the water’s surface below as it descends down the chute, (see center of image).

Once submerged, the ROV would be flown throughout the cistern, exploring any passageways and chambers. To accomplish this, pilots used video from the onboard camera and a joystick controller. An example of one such video image is shown in Fig. 2(b), where the ROV is facing back through a tight passage it once travelled. The ROV’s yellow tether can be seen feeding back into the initial chamber. Also note the water clarity in this particular cistern allowed for a reflection on the water surface, (as seen in the top half of the image).

To aid in SLAM experiments, auto depth and auto bearing controllers were used. Shown in Fig. 2(c) is one ROV using these autonomous control methods to hold station while viewing a cistern wall. It should be noted that these images were obtained by lowering two ROVs down through two

different chutes that lead to the same cistern. The second ROV was used to obtain the image of the first ROV in Fig. 2(c).

After video images of the cistern were recorded, stationary sonar scans were obtained. Each scan was taken while the ROV was sitting on the bottom of the cistern. For each scan, the ROV was positioned to ensure that scans would overlap each other to facilitate easy mosaicing.



(a)



(b)



(c)

Figure 2. For each site, the ROV was initially lowered down a deep narrow chute, (a). In (b), an image obtained while returning through a tight passage. Shown in (c) is the view from one ROV while it records images of another ROV inspecting the cistern wall.

Once a sufficient number of stationary scans were obtained, sonar scans were recorded while the ROV was in motion. Control signals, depth and heading measurements were also recorded for use with SLAM.

IV. MAPPING AND LOCALIZATION TECHNIQUES

Four different mapping and localization techniques were used, each producing maps consistent with one another, but of different formats.

A. Sonar Image Mosaics with Stationary Scans

The first approach taken was to mosaic several overlapping 360 degree sonar scans. Figure 3 displays an example mosaic created from six scans. Each scan on the mosaic has an obvious circle of high-strength returns indicating the robot's position within the scan. Note the high quality of the images and obvious correspondence between them allows for them to be easily fused by a human operator.

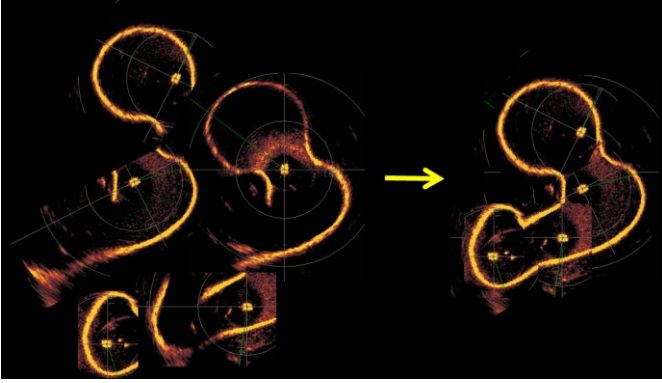


Figure 3. On the left are a collection of sonar scans obtained from site 8, a private home in Mdina Fortress of Malta. On the right is the mosaic created from the scans.

B. SLAM with the ROV in Motion

A goal of this project is to implement Simultaneous Localization and Mapping (SLAM) in real time. Since very little was known about the cisterns under investigation (i.e. size, types of features, number of features, etc.), an occupancy grid was used to represent the belief state of the environment [4]. That is, the cistern model was discretized into square cells of equal size. Each cell is assigned a probability that it is occupied (e.g. by a wall). Figure 4 shows an occupancy grid map for site 8. Note the height of the cell indicates probability of occupation.

The particular SLAM algorithm used in this project was FastSLAM for learning occupancy grids [4] since it doesn't require features like most SLAM algorithms. FastSLAM is a particle filter based approach to SLAM, in which a collection of M particles denoted as X_t is used to model the belief state. For this case, the k^{th} particle consists of an occupancy grid m_t , the robot's state $x_t^k = [x^k \ y^k \ z^k \ \theta^k \ \dot{x}^k \ \dot{y}^k \ \dot{z}^k \ \dot{\theta}^k]_t$, and a weight w_t^k that represents the likelihood that particle k represents the true state. As shown in Table 1, the t^{th} time step of the algorithm updates all particles as new sensor measurements z_t are observed.

The three key steps to this algorithm are on line numbers 4, 5 and 6. The first, **sample_motion_model**, propagates the previous state x_{t-1}^k of the robot forward in time according to the control inputs u_t . A certain degree of randomness is added propagation, in accordance with the robot's motion model.

TABLE 1. THE FastSLAM ALGORITHM

1:	Alg. FastSLAM_occupancy_grids (X_{t-1}, u_t, z_t):
2:	$X_t' = X_t = 0$
3:	for $k = 1$ to M do
4:	$x_t^k = \text{sample_motion_model}(u_t, x_{t-1}^k)$
5:	$w_t^k = \text{measurement_model_map}(z_t, u_t, m_{t-1}^k)$
6:	$m_t^k = \text{updated_occupancy_grid}(z_t, u_t, m_{t-1}^k)$
7:	$X_t' = X_t' + \{x_t^k, m_t^k, w_t^k\}$
8:	endfor
9:	for $k = 1$ to M do
10:	draw i with probability $\sim w_t^i$
11:	add $\{x_t^i, m_t^i\}$ to X_t
12:	Endfor
13:	return X_t

When the ROV is in motion, this function uses a dynamic model $x_t^k = f(x_{t-1}^k, u_t)$, which predicts the state of the ROV given the last state and current control signals. This model is based on that developed in [8]. While the model is nonlinear, it assumes decoupling between many states. Furthermore, the model in [8] doesn't include any model of the tether's effect on dynamics.

To account for both tether snags and the ROV's motion being obstructed by collision with walls, the propagation model was modified accordingly:

$$x_t^k = f(x_{t-1}^k, u_t(1 + r_1) - \varepsilon u_t(1 + r_2)) \quad (1)$$

$$\varepsilon = \begin{cases} 0 & \text{if } r_3 < \lambda \\ 1 & \text{else} \end{cases} \quad (2)$$

In equation (1), r_1 and r_2 are normally distributed random variables. The value of ε is either 1 or 0, representing a tether snag or no tether snag respectively. This is set according to a uniformly distributed random variable r_3 , and a probability of tether snag or obstruction λ .

The next step in the algorithm invokes the **measurement_model_map** function, which calculates the weight of the k^{th} particle. At a high level, the expected sonar measurement is calculated given the robot state x_t and the map m_{t-1} . This expected sonar measurement is compared with the actual measurement z_t . If the two measurements are similar, a high weight is returned, otherwise a low weight is returned.

To quantify this similarity, we first note a sonar measurement z has the form $z = [\beta \ s^1 \ \dots \ s^B]$, where β is the direction the sonar head and s^i is the i^{th} strength of return signal measured at a distance $i/\text{maxRange}$. To determine the weight of the particle, each strength of return s^i is converted to a corresponding occupancy probability according to a log odds mapping approach [4] to yield $p_z = [p_z^1 \ \dots \ p_z^B]$. If the map's cells that correspond with the B sonar measurement locations currently have occupation probabilities $p_m = [p_m^1 \ \dots \ p_m^B]$, then the weight can be calculated using a Gaussian model as:

$$w = \sum_{i=1}^B \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(p_m - p_z)^2}{2\sigma^2}\right) \quad (3)$$

The last core function of the algorithm, **updated_occupancy_grid**, updates the map with the new sonar measurements. Each return signal strength s^i is first mapped to a position according to the robot state and sonar heading β . The occupancy of the cell that corresponds to this position is updated, again according to the log odds mapping approach [4]. In general, a high signal return strength will result in a high probability of occupancy.

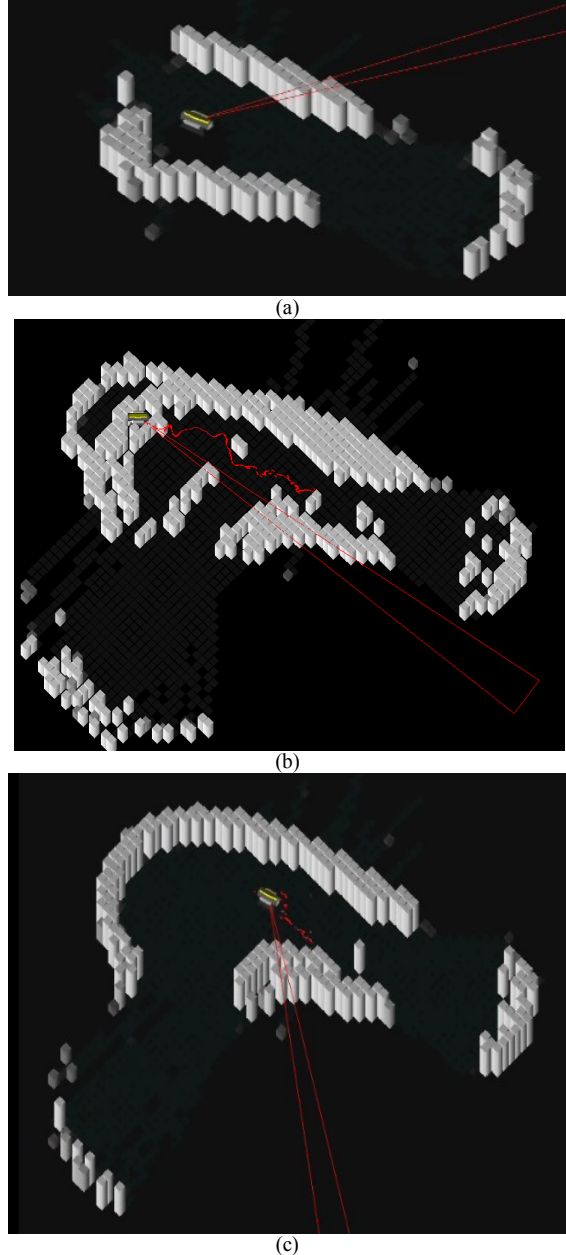


Figure 4. The ROV is mapping the cistern at site 8. In (a), the ROV sits on the bottom and maps out the mound of silt just in front of it. In (b), we see that without a model for tether snags, mapping performance is poor. Using the proposed model from equation (1), successful mapping is possible (c). The red line within the cistern indicates the path of the ROV. The two straight red lines indicate the direction of the current sonar measurement.

Lines 8 through 11 in Table 1 correspond to the resampling phase of the algorithm. In this phase, a new collection of particles X_t is generated from X_t' . That is, particles are randomly selected from X_t' and added to X_t , giving higher likelihood of selection to those particles with higher weights.

An example of the effectiveness of the SLAM implementation is shown in Figure 4. In (a), the ROV has conducted 2 sonar scans while resting motionless on the bottom. The ROV is sitting in front of a mound of sediment, resulting in a large number of strong sonar returns falsely indicating a wall just in front (i.e. just to the left of the robot in the image). With no modeling of tether or collisions, the algorithm greatly overestimates the amount of motion the ROV travels, resulting in the mapping of several walls which replicate the original wall (b). Finally, in (c) is presented mapping after several minutes of forward motion, part of which is slowed by a collision and/or tether snag. The map shows no replication of walls and appears consistent with maps produced from other methods (see Fig. 3).

C. SLAM with Stationary Sonar Scans

When using stationary scans with FastSLAM, the **sample_motion_model** function doesn't use actual control inputs. Instead the transformations (i.e. translations and rotations), required for mosaicing the stationary scans were recorded. These transformations were easy to obtain, but are subject to error. To model this error, a 2D Gaussian distribution was used, with a mean of 0 and a standard deviation σ_m . The value for σ_m was set according to the variation in transformations. Specifically, the operator transformed several sonar scan images, each 10 times, to fit within the mosaic. The standard deviations of each scan's x and y coordinates after the transformation were calculated. Of the scans transformed, the maximum value of $\sigma_m = 0.020\text{m}$ was obtained.

Figure 5 shows a map constructed using the SLAM algorithm with stationary sonar scans. Note the blue square which represents the initial scan position, located directly below site 8's access chute. Also note the ROV's final scan position in the bottom of the image.

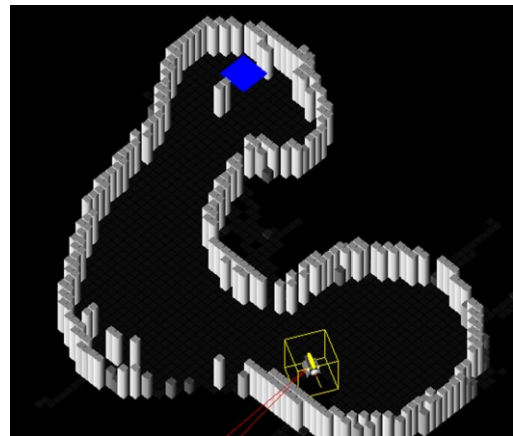


Figure 5. An example of a map created by inputting static sonar scans into a SLAM algorithm. The ROV in the image indicates the state of the ROV during the final sonar scan. The blue square indicates the approximate position of the cistern access point.

D. Localization Using Previously Constructed Maps

Once maps are constructed using any of the previous techniques, the robot can navigate using a localization algorithm to estimate the robot's state within the map. In this work, Particle Filter Localization was implemented [4]. The algorithm was similar to the FastSLAM algorithm presented in Table 1, with step 6 removed to leave the map unchanged over time.

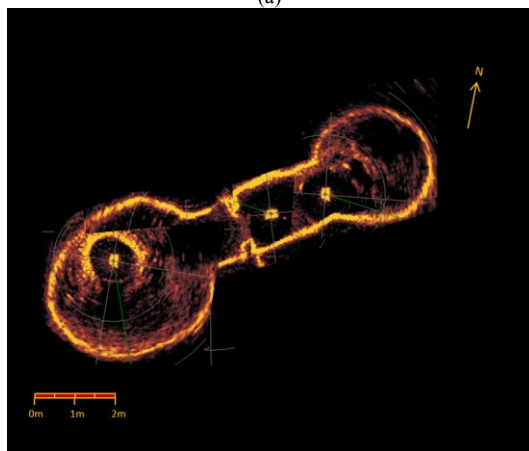
V. RESULTS

Six of the eight 8 sites visited had a sufficient water depth and were mapped, providing new and useful information for archaeological purposes. However, different levels of success were achieved depending on the method used and the site in question.

The mosaics created for all sites provided information regarding the orientation, scale, and complexity of the cisterns. Figure 6 shows examples from two sites. As can be seen in (a), a small chamber which lies at the bottom of the access point to site 2 is connected to a larger reservoir. This was observed in 3 of the 6 sites.



(a)



(b)

Figure 6. Examples of sonar mosaics created using stationary sonar scans. A map of the cistern in site 2 (Gozo Citadel) is shown in (a). In (b), a map of the cistern from site 6 (Private home in Mdina) is shown.

In Fig. 6(b), a tight passage connects two bulb-shaped chambers. The north-east chamber lies at the bottom of the access point of site 6. Upon visual inspection using video camera, another access point (although covered) was found to be above the south-west chamber.

In validating the SLAM while-in-motion approach to mapping cisterns, data was only obtained for 2 of the sites. A significant issue that limited data was the inability to drive the ROV with complete control when running the on-line SLAM algorithm. When the algorithm is running, the ROV must be controlled via computer interface which was not a problem in previous pool trials. However, navigating narrow passageways required more sensitive control similar to that provided by the ROV's original joystick control.

Despite these difficulties, it has been shown that the algorithm works well in mapping the cisterns. Figure 7 shows an occupancy grid map created for site 8, (accessed from a private home in Mdina). In this example, only 25 particles were used.



Figure 7. Example of a map created from implementing FastSLAM while in motion.

Using static sonar scans within the SLAM algorithm proved effective in mapping the cisterns. Figure 8 shows a map of the cistern at site 8.

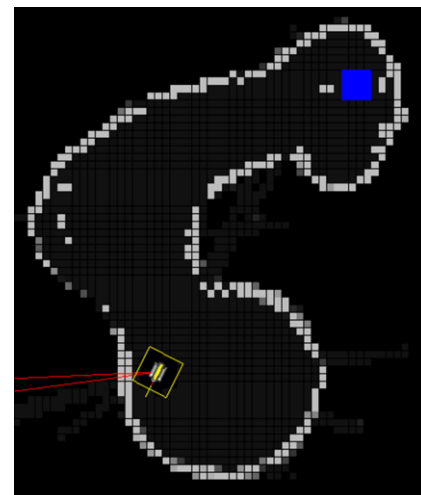


Figure 8. Example of a map created from implementing FastSLAM with static sonar scans.

Finally, the Particle Filter implementation showed positive results in that the robot always converged to within 0.5m of the actual location, despite having no knowledge of the initial state. An example is provided in Figure 9. In (a), 500 particles are shown that each represents a possible state of the robot. The robot's state estimate is calculated as the weighted average of all particle states, and is shown in the center of the image. The actual position is shown as a blue square. Despite the fact that the robot has not moved, it can localize itself with only 2 scans of the area, as shown in (b). Figure 9(c) shows the localization error as a function of time.

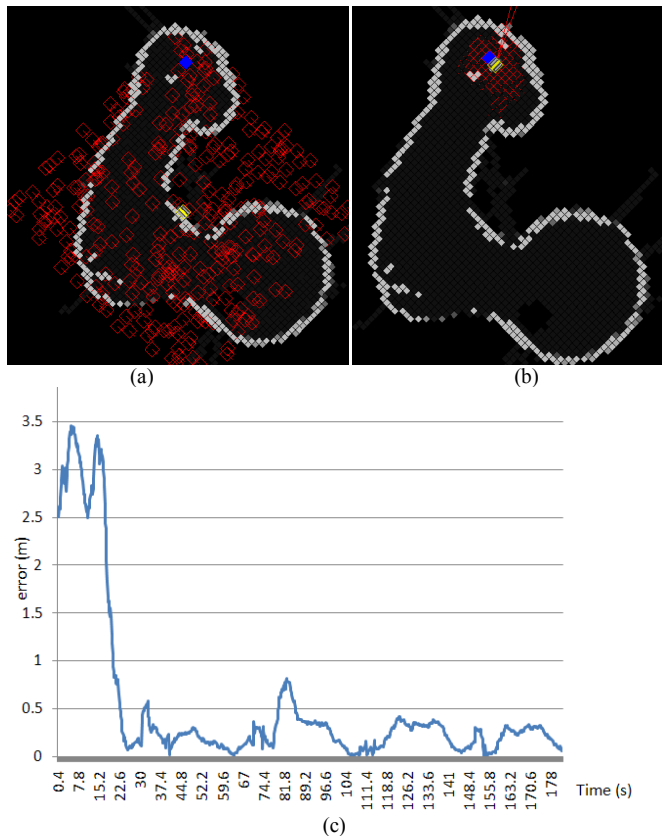


Figure 9. An example of Particle Filter localization being conducted with data from site 8. Initially, the robot has no idea where it is located, as shown by the randomly distributed set of red particles (a). After a few sonar scans (b), the robot can successfully localize itself with respect to the actual position (blue square). In (c), the error in position is plotted.

VI. CONCLUSIONS & FUTURE WORK

The first cistern mapping expedition in Malta and Gozo successfully constructed maps for use in archaeology study of these ancient water storage systems. In each cistern, a small ROV was deployed which collected sonar data from various positions in the cistern. Using these data sets, four methods for mapping and localization were investigated. Stationary scan methods, including scan mosaicing and FastSLAM, worked well. Implementing FastSLAM while moving had success but was validated by only a few data sets. Particle Filter localization also worked very well in that state estimates converged to actual states despite there being no knowledge of initial conditions.

In the future, scalability of the FastSLAM implementation will be improved. Also, computer control will be fine tuned, allowing precision control of the ROV in narrow passages.

ACKNOWLEDGMENT

This work would not be possible without funding and resources made available by the AURORA Special Purpose Trust and the California State Faculty Support Grant Fund. A special thanks to George Azzopardi, Godwin Vella, Dun Guzepp, and owners of private homes in Mdina who provided access to their properties. Other supporters included Fondazzjoni Patrimonju Malti, Heritage Malta, VideoRay LLC, and Trittech International Limited.

REFERENCES

- [1] H.P. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, Vol 9, pp. 61-74, 1988.
- [2] K. Richmond and S. M. Rock, "An operational real-time large-scale visual mosaicing and navigation system," *Proceedings of the 2006 MTS/IEEE OCEANS conference*, September 2006.
- [3] H. Sakai, T. Tanaka, T. Mori, S. Ohata, K. Ishii, and T. Ura, "Underwater video mosaicing using AUV and its application to vehicle navigation," *Proceedings of the 2004 International Symposium on Underwater Technology*, April 2004.
- [4] S. Thrun, W. Burgard and D. Fox, "Probabilistic Robotics," MIT Press, 2005.
- [5] C. Baker, A.C. Morris, D. Ferguson, S. Thayer, C. Whittaker, Z. Omohundro, C. Reverte, W.L. Whittaker, D. Haehnel, and S. Thrun, "A Campaign in Autonomous Mine Mapping," *Proceedings of the IEEE Conference on Robotics and Automation (ICRA)*, April 2004.
- [6] S. B. Williams, P. Newman, G. Dissanayake, H. Durrant-Whyte, "Autonomous underwater simultaneous localisation and map building," *Proceedings of the 2000 IEEE International Conference*, 2000.
- [7] N. Fairfield, G. Kantor, and D. Wettergreen, "Real-time SLAM with octree evidence grids for exploration in underwater tunnels," *Journal of Field Robotics*, Vol 24, Issue 1-2, pp. 03-21, October 2006.
- [8] W. Wang, and C. M. Clark, "Modeling and simulation of the VideoRay Pro III underwater vehicle," *Proceedings of the IEEE OCEANS'06 Asia Pacific IEEE Conference*, May 2006.