Universitat de Girona

UNDERWATER SLAM FOR STRUCTURED ENVIRONMENTS USING AND IMAGING SONAR

David RIBAS ROMAGÓS

ISBN: 978-84-691-8357-1
Dipòsit legal: Gi-1471-2008

Universitat de Girona

Department of Computer Engineering

PhD Thesis

Underwater SLAM for Structured Environments Using an Imaging Sonar

Thesis presented by
David Ribas Romagós,

to obtain the degree of:
PhD in Industrial Engineering.

Supervisors:
Dr. Pere Ridao Rodríguez
Dr. José Neira

Girona, May 2008

For Cristina and my parents.

Abstract

Underwater SLAM for Structured Environments Using an Imaging Sonar

This thesis is concerned with the navigation problem for autonomous underwater vehicles operating in artificial structured environments like harbours, marinas, marine platforms and other similar scenarios. Obtaining an accurate position in such scenarios would notably increase the capabilities of underwater vehicles and open the door to real autonomous operation. Maintenance, inspection and surveillance of marine installations are only a few examples of possible applications.

The principal contributions of this thesis consist of the development of different localization systems for those situations in which an *a priori* map of the environment is available but, in particular, in the development of a novel solution to the Simultaneous Localization and Mapping (SLAM) problem. This solution pursues the objective of providing an autonomous vehicle with the ability to build a map within an unknown environment while, at the same time, using this same map to keep track of its current position.

A mechanical scanning imaging sonar has been chosen as the principal sensor for this work because of its relative low cost and its capacity to produce a rich representation of the environment. On the other hand, the particularities of its operation and, especially, its low scanning rate, have presented many challenges during the development of this proposed localization strategies. The solutions adopted to address these problems constitute another contribution in this thesis.

The development of underwater vehicles and their use as experimental platforms is another important aspect of the research work presented here. Experiments carried out in both laboratory and real application environments have provided the different datasets necessary for the testing and evaluation of the different localization approaches.

Acknowledgements

I would like to express my gratitude to the people who have supported me during the elaboration of this PhD thesis. First and foremost to my two advisors; Pere Ridao, for believing in me and encouraging me during this research with his unshakable optimism, inspiration and friendship, and José Neira, for introducing me to the secrets of SLAM during my stay at the University of Zaragoza and for giving me innumerable helpful comments and suggestions.

I also thank the members of the Underwater Robotics Lab for their camaraderie and willingness to help whenever it was necessary. I want to thank my colleagues Marc, Andrés, Narcis and Emili for always being there, and also our recent incorporations in the lab; Sik, Tali, Guillem, Xavi, Aggelos, Tomeu and Enric. I would also like to extend my gratitude to the rest of the members of the Computer Vision and Robotics Group and of the department of Computer Engineering, in particular to Jordi F., Xevi C., Rafa, Miki, Joan B., Pep, Quim, Nuno, Carles M., Muñoz, Tudor, Quintana, Marina, Olivier, Armagan, Robert, PPL1, Lluis, Tomas, Toni, Ingrid, Anna, Silvia, Maria and Rosa. Further, I should not forget the people who have passed though the lab/department during these years, Javier A., Toni "Senglar", Josep, Eduard, Marc, Ela, Anna, David, Francois, Mangui, Bet, Jordi, Jonathan, Isela, Montse, Marta, Tesh, Marc R., Marc C., Isidro, Revenga, Yoyas, Joan E. and Carles.

I am also grateful to the people in the Robotics and Real Time Group at the University of Zaragoza for their warm reception and hospitality during my stay. I especially thank Juan D. Tardós for his insightful comments and revisions of my work, and also to Ana, Dario, Lina, Pedro, Ruben, Alejandro, Javier, Iñaki, Luis, Diego and Eduardo, for making me feel like one more of the group. I should not forget Antoni, from the university of the Balearic Islands, with whom I coincided during my stay and shared great moments.

I cannot finish these acknowledgements without mentioning my beloved parents, Josep and Concepció, and brother, Jordi. They deserve special gratitude for their comprehension and support of my decision to start a

research career. I wish to extend my gratitude to all my friends, but in particular to Dani and Maria, and also to Tomas, Edu, Manolo, Jaume, Felipe and Pum for reminding me that there is life outside the lab.

The last acknowledgement is reserved for the most important person in my life, Cristina, who with her love and encouragement, has been my principal support during these years.

Contents

List of Figures

List of Tables

List of Algorithms

Glossary

AUV	Autonomous Underwater Vehicle
CIRS	Centre d'Investigació en Robòtica Submarina
CML	Concurrent Mapping and Localization
CPE-SLAM	Constant Pose Estimation SLAM
CSS	Co-operating Statistical Snake
CTS	Constant Time SLAM
DCRO	Deferred Reference Counting Octree
DGPS	Differential Global Positioning System
DOF	Degree Of Freedom
DSM	Decoupled Stochastic Mapping
DVL	Doppler Velocity Log
EIF	Extended Information Filter
EKF	Extended Kalman Filter
FOG	Fiber Optic Gyro
GESMA	Groupe d'Etudes Sous-Marines de l'Atlantique
GPF	Geometrical Projection Filter
GPS	Global Positioning System
HRA	High Resolution Array
IC	Individual Compatibility
INS	Inertial Navigation System
JC	Joint Compatibility
LS	Least Squares
MRF	Markov Random Field
MRU	Motion Reference Unit
MSIS	Mechanical scanning imaging sonar
NN	Nearest Neighbor
NUWC	Naval Undersea Warfare Center
RLG	Ring Laser Gyro
RANSAC	RANdom SAmple Consensus
ROV	Remotely Operated Vehicle

RTS	Rauch-Tung-Striebel
SAS	Synthetic Aperture Sonar
SAUC-E	Student Autonomous Underwater Challenge - Europe
SBL	Short Base Line
SLAM	Simultaneous Localization And Mapping
TMS	Tether Management System
USBL	Ultra Short Baseline
UUV	Unmanned Underwater Vehicle

Chapter 1

Introduction

More than 70% of the earth's surface is covered by water. Oceans and seas host an incredibly rich biodiversity, influence short and long term climate and have a high impact, not only on economy, but on the whole life and evolution of human society. Exploring this large body of water is a matter of the utmost importance, not only because it represents a vast source of natural resources, but also because its study may help us understand how this complex ecosystem works.

Remotely Operated Vehicles (ROVs) have been one of the best tools to undertake this mission, making the exploration of the deepest regions possible while avoiding risk to human lives. The first ROVs were developed in the 1960s to perform rescue and recovery operations, the technology however, soon evolved to other uses. Many applications of ROV technology can be found in the oil and gas industries, where it is not uncommon to find tasks exceeding the reach of human divers. In fact, ROVs have become an essential tool without which the explotation of offshore oil fields would not have been possible. Their missions range from the surveying of operation areas to the deployment, inspection and mantainance of undersea structures like oil rigs or pipelines [Iovenittl et al., 1994]. These vehicles have also played an important role in the scientific community, making possible the study of the ocean possible in many different ways. Now the sea floor can be mapped with bathymetric sensors while image mosaicking techniques make the generation of large visual maps possible [Ferrer et al., 2007]. Moreover, zones of geologic or biologic interest can be explored and sampled, from the oceanic ridges to the deepest trenches. Even archeologists and historians now have access to sunken remains and shipwrecks [Eustice et al., 2005]. ROVs can also be found in military missions, cleaning the path of ships in mine hunting operations or assisting in inspection or salvage tasks.

Missions involving the use of ROVs are complex. These vehicles are gen-

erally linked to a ship on the surface by means of an umbilical cable or *tether* which provides power and communications. In this way, operators on the ship can control the vehicle and receive the feedback from the onboard sensors. Although this link makes operating ROVs for a long time period possible, the infrastructure requirements are high as are the costs. Campaigns may take place far from the coast, which involves using a ship of considerable dimensions with its corresponding crew and specialized ROV operators for many weeks. Furthermore, the ship has to be equipped with a crane for the deployment and recovery operation, as well as with other elements like sufficient a tether for the desired operational depth and even a Tether Management System (TMS) for those operations at greater depths.

In recent years, efforts have been made to provide those vehicles with a greater degree of autonomy. The objective is to remove the link with the surface ship to expand the vehicle's capabilities and, at the same time, reduce the operational costs. This is achieved by equipping the submersible with its own power source, generally batteries, and giving it the capacity to determine its actions based on the inputs from its own sensors and a pre-defined mission plan. The result of this research are the so called Autonomous Underwater Vehicles (AUVs) which nowadays have already succeed in performing different types of tasks, in particular, those related with the collection of sensor data or the production of detailed maps of portions of the seafloor [Petillot et al., 2002]. There are also a few designs for AUVs capable of performing undersea intervention operations. However, this research is still in its early stage and those vehicles cannot be considered as operational.

The development of AUVs has offered numerous advantages, but has also presented new challenges. One of the most significant examples is the problem of underwater navigation or, in other words, how to determine the vehicle's position within the environment so it can take the correct actions to successfully accomplish the mission. Traditionally, the problem has been addressed in various ways, none of which being fully adequate. The positional error growth associated with dead reckoning based on Inertial Navigation Systems (INS) and/or Doppler Velocity Logs (DVL) make their use impractical for long term navigation [Kuritsky and Goldstein, 1990; Paglia and Wyman, 1996]. In order to avoid this problem, data from a Global Positioning System (GPS) receiver can be used to provide navigation resets but, due to the null coverage in underwater environments, it can only be effectuated when on the surface. Alternatively, artificial beacons may be employed for long term underwater positioning. Many configurations are available for these beacon systems, such as Long Base Line (LBL), Short Base Line (SBL) or Ultra Short Baseline (USBL) (see Figure 1.1) [Heckman and Abbot, 1973; Hunt et al., 1974; Milne, 1983]. Unfortunately, there are numerous missions

in which these solutions are unfeasible. The need of prior beacon deployment, the high cost and the constraints in the working volume of the AUV are the principal disadvantages. However, there are other ways to achieve localization of an AUV without the need of external hardware. Map matching techniques use information from onboard sensors to provide ground-fixed, feature-relative localization given an *a priori* map of the environment [Lucido et al., 1996]. Variants of the method can be found for gravitational anomaly and magnetic field maps [May, 1978; Tyren, 1982]. Even so, its main drawback is that an up-to-date map of sufficient resolution will not be available for many operating areas. So, again, a self-contained system would be preferable , with no need of previous knowledge of the terrain or external devices to obtain a reliable localization of the vehicle.

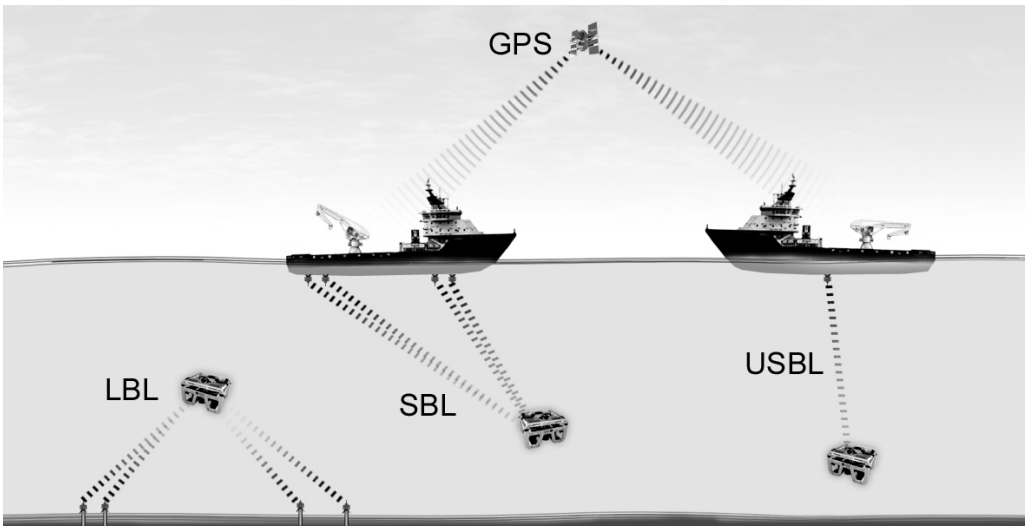


Figure 1.1: Typical configurations for AUV localization using LBL, SBL and USBL.

The Simultaneous Mapping and Localization (SLAM), a research topic which has attracted a great deal of attention in the research community for almost two decades, may be the solution to the navigation problem. It proposes a method for autonomous vehicles to build a map within an unknown environment while keeping track of their current position. This process is carried out using only the vehicle's sensors for the perception of the surrounding environment and/or its own state and hence, can be considered as a self-contained system. This thesis is concerned with the application of SLAM techniques to the field of autonomous underwater navigation. Next,

some background on the motivations and applicability will be provided, as well as a description of the thesis' objectives and the outline of this document.

1.1 Antecedents

The research presented in this thesis has been carried out in the Underwater Robotics Laboratory of the Computer Vision and Robotics Group of the University of Girona. This group has been doing research in underwater robotics since 1992. The main contribution over the past few years is the development of three Unmanned Underwater Vehicles (UUV). The first prototype, called GARBI, was developed in collaboration with the Polytechnic University of Catalonia. This vehicle was initially conceived as a ROV, but after successive modifications over the years, the vehicle evolved into its final configuration as an AUV in 2005. The second prototype, URIS (1999), was fully developed at the University of Girona and was designed as an small AUV for testing in laboratory conditions. The most recent vehicle is the Ictineu (2006), an AUV which brings together the broad sensorial capabilities of the GARBI and the small form factor of the URIS, which make this vehicle a perfect research platform for testing in both laboratory and real application environments.

The research efforts in the Underwater Robotics Laboratory have been oriented to the development of the diverse disciplines related with the operation of autonomous vehicles. An example can be found in the work done in control architectures, which has lead to the creation of the O^2CA^2 control architecture [Carreras et al., 2005], but also in the recent advances made towards a control mission system [Carreras et al., 2007]. Parallel work has also been done in the identification of the dynamic/kinematic models of the vehicles [Ridao et al., 2004] which has made the development of research tools such as the Neptune simulator possible [Ridao et al., 2004b]. With respect to the application domain, preliminary work was carried out on the use of ROV technology for inspection of hydroelectric dams using image mosaics [Batlle et al., 2003]. Later, the topic was readdressed using AUVs in the context of a research project supported by the Spanish commission MYCT, made in collaboration with the University of the Balearic Islands and the Polytechnic University of Catalonia. The objective of the project was to develop the capacity of AUVs for their use in industrial applications such as the inspection of hydroelectric dams, harbours and underwater cables and pipes.

According to the achievements in those research lines and application domains, the capabilities of our vehicles have increased as well as the navigation requirements. The work done during the elaboration of this thesis has contributed to the beginning of a new research line whose objective is

to improve the navigation capabilities of our vehicles. Our knowledge of this new topic has been complemented with a collaboration with the Robotics and Real Time group of the University of Zaragoza. This group has had long experience in sensor perception and navigation systems and has been a referent in SLAM research since its origins with numerous relevant contributions to the field. A consequence of this relationship is the co-tutoring of this thesis.

1.2 Goal of the Thesis

As mentioned in the antecedents, the goal of this thesis is the study and development of navigation systems for AUVs, with special attention to the application of SLAM techniques as a self-contained system which requires neither previous knowledge of the scenario nor the use of absolute positioning systems like GPS, LBL or USBL. Moreover, and in coherence with the presented application domains, the system will be designed for use in structured environments like those present in many industrial scenarios, specifically for scenarios containing manmade structures in the form of rectilinear walls like those met in harbours, breakwaters, marinas, canal systems, etc. Although most of the previous work done in this field focuses on open sea and coastal applications, obtaining an accurate positioning in such scenarios would increase AUVs capabilities notably. For instance, an AUV could use a harbour as an outpost for oceanography research if it is able to localize itself and navigate with enough accuracy to safely perform the leaving and returning operations [Griffiths et al., 1998]. Maintenance and inspection of underwater structures [Kondo et al., 2006; Martins et al., 1999] and even surveillance of marine installations are examples of other applications that can benefit from such a system.

Focusing on such scenarios will offer several advantages. For instance, one of the most critical issues when operating in underwater environments is the shortage of reliable landmarks to use in the map. Although the quantity of landmarks will depend on each application scenario, walls usually produce strong sonar returns which are much more constant and reliable than natural targets. Moreover, walls are usually vertical structures and therefore, a planar map will be sufficient in most cases.

1.2.1 Objectives

After reviewing the research antecedents and describing the problem, the goal of this thesis is stated. The general purpose is summarized as:

"The development of a SLAM approach for an AUV to achieve localization in man-made structured underwater environments using a mechanically scanned imaging sonar as principal sensor"

The term "man-made structured environment" should be understood as an environment containing by artificial, previously existing structures characteristic to the scenarios for the applications at hand and where no additional elements have been introduced to serve as landmarks for the SLAM framework. This goal was selected in order to start a new research line on autonomous navigation, but also as the next logical step given the antecedents of the group and the research projects in which it is involved. Moreover, SLAM is one of the most active topics in robotics research and, although a large number of works have already been presented, there are still very few approaches applied to the field of underwater robotics. The application of SLAM algorithms in AUVs was certainly the most important purpose of this dissertation. For this reason, it was a priority to perform the experimentation with real data from a robot in the objective scenario as a premise to demonstrate the achieved research advances.

The goal of this thesis can be divided into the following more specific objectives:

Underwater localization with an *a priori* map. Exploring and designing different approaches to perform underwater localization for an AUV equipped with mechanically scanned imaging sonar operating in a structured scenario whose map is previously known. The interest in this topic is focused on exploring the use of Kalman filters and data association algorithms for underwater localization.

Feature extraction. Development of a feature extraction method capable of dealing with the particular complexities of a mechanically scanned sonar. This method should be able to detect the presence of walls in the environment as line features in the acoustic data, as well as to estimate the uncertainty of this observation. The detection of the features will rely on the use of a robust data association algorithm, while the estimate of the observed feature will be based on the imprint left by the landmark in the acoustic data.

Underwater SLAM. Develop a SLAM framework based on the stochastic map approach [Smith et al., 1990] using the commented feature extraction method and the experience gained during the development of the localization techniques. The purpose of this SLAM system is to

localize a vehicle within a structured environment typical in previously commented applications.

Experimentation with an AUV. Evaluation of the proposed localization and SLAM systems with real experiments using an AUV. The feasibility and limitations of these approaches must be experimentally tested with the available systems and resources.

1.3 Outline of the Thesis

The contents of this thesis can be divided into three parts. The first part overviews the SLAM problem (Chapter 2), paying special attention to its application in underwater environments, introduces the Ictineu AUV (Chapter 3), a research vehicle developed during the elaboration of this thesis, and describes the operating principles of a mechanically scanned imaging sonar as well as the issues related with their application in the problem (Chapter 4). The second part presents three different localization algorithms (Chapter 5) as introduction to some of the techniques and solutions employed later in the proposed SLAM algorithm (Chapter 6). Experimental results endorse the different proposals. Finally, the last part of the document summarizes the contributions and comments on further work (Chapter 7). A brief description of each chapter is presented next.

Chapter 2: *State of the Art.* This chapter presents the field of "Simultaneous Localization and Mapping", relates the history and development of the problem and overviews the most remarkable works performed on underwater environments. A discussion and the aims of this work can be found at the end of the chapter.

Chapter 3: *The Ictineu Autonomous Underwater Vehicle.* This chapter introduces the Ictineu AUV, the research platform developed during the elaboration of this thesis which was used to test the proposed SLAM framework. The main features of the vehicle are described, including many mechanical aspects, the equipped sensors and its software architecture.

Chaper 4: *Understanding Mechanically Scanned Imaging Sonars.* This chapter describes the basic operating principles of the sonar sensor which was chosen to perform SLAM. Some hints are provided about the interpretation of sonar images. Moreover, many issues that need to be addressed before using the sensor are identified.

Chapter 5: *Localization with an a priori Map.* This chapter describes three different approaches for localization with previous knowledge of the scenario. This chapter should be understood as an exploration of some of the techniques that will then be developed in the SLAM approach. Experimental results are included for each of the presented methods.

Chapter 6: *Simultaneous Localization and Mapping.* This chapter proposes a SLAM algorithm for AUVs equipped with a mechanically scanned imaging sonar operating in manmade environments. Details are provided about a novel method for extracting line features and their uncertainty from acoustic images as well as about the implemented Kalman filter framework. At the end of the chapter, experimental results including a SLAM executed with real data obtained in an application scenario with the Ictineu AUV are presented.

Chapter 7: *Conclusion.* This chapter concludes the thesis by summarizing the work and points out contributions and future work. It also comments on the research evolution and the publications accomplished during this research project.

Appendices: This chapter incorporates additional information on some of the topics introduced in the thesis.

Chapter 2

State of the Art

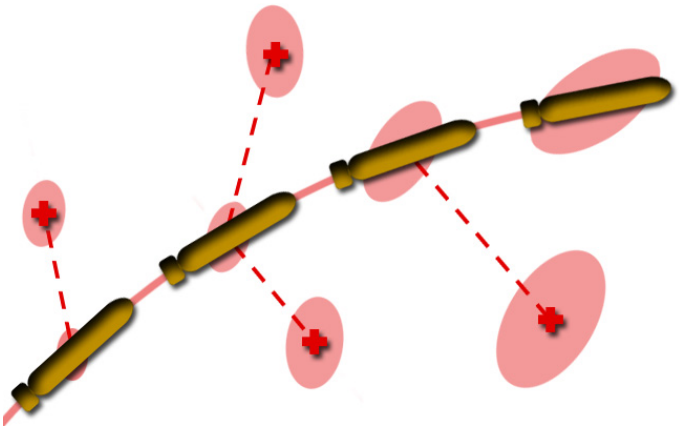
2.1 The SLAM problem

Simultaneous Localization and Mapping (SLAM), also referred to as Concurrent Mapping and Localization (CML), is a fundamental problem in mobile robotics that has been the focus of an important amount of research work in recent years [Durrant-Whyte and Bailey, 2006; Bailey and Durrant-Whyte, 2006]. The objective of SLAM is to make it possible for a moving robot starting in an unknown location without previous knowledge of the environment, to build a map using its onboard sensors while, at the same time, using this same map to compute the robot's location. Although performing these two tasks simultaneously may seem complex, the essentials behind SLAM are indeed quite simple. Figure 2.1 illustrates the basics of the process. A moving vehicle will inherently accumulate errors in its position estimate as a consequence of the noise introduced in the dead-reckoning and/or the inaccuracies in the use of a prediction model. Moreover, errors will also affect the map building process. The sensor that perceives the environment is mounted in the vehicle and therefore, its position uncertainty will be incorporated when new information is added to the map. As a result, the vehicle will eventually get lost and the map will become unusable (Figure 2.1(a)). A system performing SLAM however, is able to attenuate and even contain this uncertainty growth by means of the reiterate observation of the elements stored in the map. Figure 2.1(b) represents the situation where a new measurement from the robot is susceptible to correspond to an entity already incorporated in the map. Then, a data association process should be carried out to determine the correct matching. When this process is positive, this information is used to update the estimates of both the vehicle's position and the map. Adding more information results in a better estimate and hence, a reduction

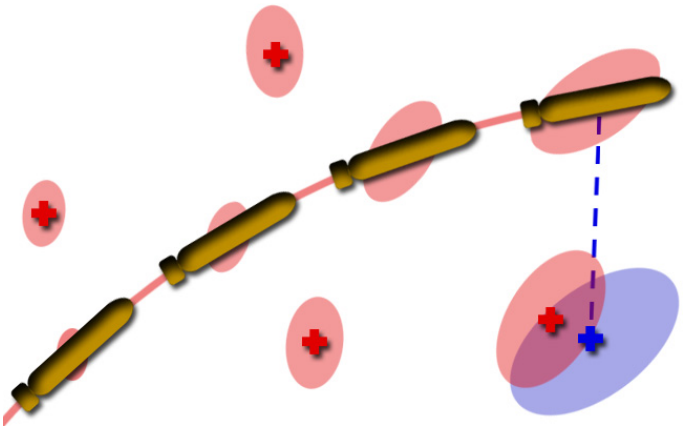
of the uncertainty in the problem (Figure 2.1(c)).

2.2 History

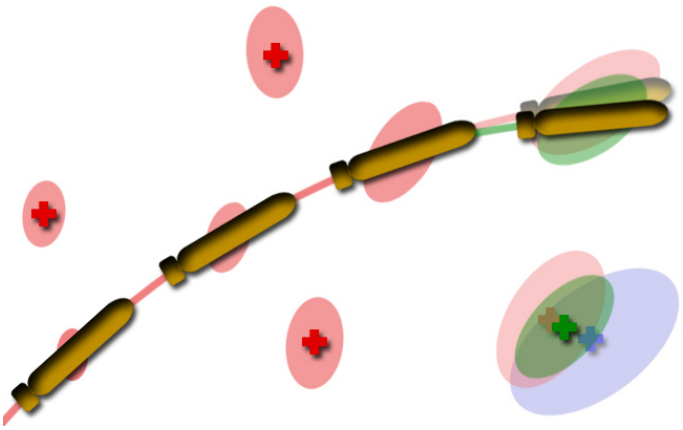
In the 1980s and early 1990s, navigation and mapping were treated as separate problems. The work done in the field of mapping was widely divided into topological [Kuipers and Byun, 1991; Choset, 1996] and metric approaches, using either occupancy grids [Elfes, 1987; Moravec, 1988] or a geometric description of the environment [Chatila and Laumond, 1985]. On the other hand, the work done on localization explored map/scan matching techniques [Lu and Milios, 1997; Cox, 1991] and the use of geometrical landmaks in Kalman filters [Leonard and Durrant-Whyte, 1991]. The success in those strands of research created the proper environment for the birth and development of the SLAM problem. According to Durrant-Whyte and Bailey [Durrant-Whyte and Bailey, 2006], its origins can be dated back to 1986, when many discussions about the application of estimation theories to mapping and localization problems were carried out by a number of researchers during the IEEE Robotics and Automation Conference held in San Francisco, California. Over the next few years, the seminal works of Smith and Cheeseman [Smith and Cheeseman, 1986] and Durrant-Whyte [Durrant-Whyte, 1988] were presented, setting the statistical basis for the description of spatial relationships between landmarks and the manipulation of geometric uncertainty. In was not, however, until some years later that the developments in this new research line culminated in the publication of the notable paper by Smith et al. [Smith et al., 1990]. This paper showed that a consistent full solution of the SLAM problem would require a joint state containing the position of the vehicle and the features in the map to represent all the correlations that appear as a consequence of vehicle error which is common to all the relative landmark measurements obtained during the creation of the map. This implies that a map containing a large number of landmarks will require a huge state vector and hence, a considerable computational cost to perform the estimation. This work, however, did not analyze the convergence properties of the problem. In fact, at the time, it was generally assumed that the map errors would not converge but would grow without bound. These two reasons, the computational complexity and the lack of knowledge about convergence, made the SLAM research come to standstill and as a result, the efforts were again focused on dealing with mapping and localization as separate problems. It was not until the later 1990s, when the convergence properties where finally elucidated [Durrant-Whyte et al.,



(a) Uncertainty growth in the map landmarks and vehicle locations.



(b) The new measurement (blue) is a re-observation of a landmark in the map.



(c) Incorporating the new information results in a better estimate (green) of both map and vehicle positions.

Figure 2.1: The SLAM problem.

1995; Csorba, 1997]. The correlations between landmark estimates increase monotonically with the number of observations made, which means that the knowledge of their relative positions improves and never diverges and, hence, a better map is obtained regardless of the vehicle's errors [Dissanayake et al., 2001]. This breakthrough revived interest in the SLAM problem. In fact, over the last decade this field has experienced a substantial expansion and researchers have focused on many different lines of research from dealing with computational complexity [Guivant and Nebot, 2001; Leonard and Feder, 2000] to data association techniques [Neira and Tardós, 2001]. Different solutions to the probabilistic SLAM problem have also been proposed as alternatives to the traditional implementations of the stochastic map with extended Kalman filters (EKF) [Thrun et al., 2005]. Some other efficient strategies using Gaussian uncertainty models include postponement [Knight et al., 2001], decoupled stochastic mapping [Leonard and Feder, 2001], the compressed filter [Guivant and Nebot, 2001], sequential map joining [Tardós et al., 2002] and the constrained local submap filter [Williams et al., 2002]. Alternatively, other implementations such as Information Filters and particularly, its non-linear version, the extended information filter (EIF), have been used recently in order to reduce the computational cost [Thrun et al., 2004]. Like the EKF, the EIF represents the uncertainty with a Gaussian. However, its main difference is the use of an alternative parametric representation to characterize the belief, which leads to slightly different equations and approximately sparse matrixes that offer a better computational efficiency. In spite of their popularity, the convergence of systems modeled under a Gaussianity assumption have only been demonstrated for the linear case, while non-linear systems have shown to be inconsistent as a consequence of linearization errors [Castellanos et al., 2004]. The issues related with the representation of non-Gaussian probability distributions have been addressed with the use of particle filters [Thrun, 2001; Montemerlo et al., 2002]. This technique uses a finite number of sample states drawn from the estimate, called particles, to represent the uncertainty distribution. The greater the number of particles, the better the description of the uncertainty. Luckily, the number of particles can be adapted to the suspected complexity of the estimate in order to obtain computationally efficient algorithms. The FastSLAM, introduced in [Montemerlo et al., 2002], is one of the most remarkable implementations of this typology of probabilistic filters. In contrast to feature based techniques, the Constant Pose Estimation (CPE) SLAM [Lu and Milios, 1997; Gutmann and Konolige, 1999] is a method that makes use of dense sensor data, maintaining a network of local constraints between the robot's positions and producing the map through optimization. Its main advantage is that such a representation scales well with the map area because it gen-

erally represents only the local constraints. In parallel to the evolution of the probabilistic methodologies, the domains of application have also experienced a significant expansion. They have not been limited only to dealing with indoor environments of increasing complexity [Castellanos et al., 1999; Newman et al., 2002; Bosse et al., 2003; Davison et al., 2004], but also SLAM systems have been successfully deployed to work in challenging outdoor scenarios [Guivant and Nebot, 2001; Clemente et al., 2007], in the air [Kim and Sukkarieh, 2003] and even at sea [Newman and Leonard, 2003; Williams et al., 2001; Tena et al., 2003b]. These achievements evidence how the intense research has lead to the definition and understanding of the main working principles of the SLAM problem. However, although it can be considered solved for small/medium environments, there are still some open issues: the optimization of computational burdens, consistency, data association, the definition of better map representations or the deployment of SLAM systems in new and challenging application domains are examples of the problems that will probably be intensely addressed in the near future.

2.3 The use of sonars for SLAM

Even though laser scanners are expensive, they are probably one of the most popular sensorial choices in either indoor or outdoor applications [Thrun, 2001; Newman et al., 2002]. This is mainly because they provide high quality dense data with a good angular precision. Another popular alternative is the use of one or more cameras to obtain visual information (e.g. color, shape or texture) from the environment [Davison et al., 2004; Folkesson et al., 2005; Clemente et al., 2007]. On the other hand, acoustic sensors have usually been considered one of the cheapest but less reliable sensorial options to perform SLAM. Even when operating in highly structured environments, sonars produce measurements with poor angular resolution and ghost returns appear as a result of specular reflections [Leonard and Durrant-Whyte, 1992]. Many remarkable works have dealt with these limitations. For instance, in [Leonard and Rikoski, 2000; Leonard et al., 2002], the indetermination in bearing measurements from an air sonar have been addressed by using batches of range-only data acquired from multiple vantage points. The work presented in [Tardós et al., 2002] went even beyond with the implementation of a voting procedure which made discriminating different types of features and reject spurious measurements possible. Works like these demonstrate that, despite their poor precision, air sonars can be used in SLAM, but the fact is that there are still few examples of air sonar systems performing SLAM in large areas. In underwater environments, however, the situation is

the opposite. Laser based sensors are impractical because of the attenuation and dispersion of the light in water, while, by similar reasons, the use of cameras is limited to applications where the vehicle navigates in clear water and very near to the seafloor. This leaves acoustic devices as one of the best options for underwater sensing. The excellent propagation of sound in water makes possible for an acoustic wave to travel many thousands of meters without the signal losing significant energy. So, it is not unusual to find sonars capable of measuring at long ranges even in turbid water conditions. Generally, the sonars equipping underwater vehicles are devices technologically more advanced than the ones equipping their indoor/outdoor counterparts. Sophisticated designs of the transducer heads and the use of beam forming techniques make obtaining narrow beams which can produce really precise bearing measurements possible. Moreover, in contrast with the confined spaces found in indoor applications, the open spaces in underwater scenarios usually produce more reliable data. Active sonars (i.e. sonars which can both transmit an acoustic signal and receive its reflected echo) are the most adequate for SLAM because of their capacity to extract information from the environment. They can be classified in two categories depending on whether they produce only a set of range and bearing measurements or an acoustic image of the scene. Among those in the first category, the most commonly used are (Figure 2.2):

Echo sounder: This is one of the simplest and least expensive systems for measuring range. The echo sounder operates by emitting a pulse from its transducer. When this pulse reflects off a surface, it returns to the sensor head and the time of flight can be measured and therefore, the distance estimated. These kinds of devices are usually mounted in a down-looking position to find the altitude of the vehicle with respect to the seabed.

Mechanically scanned profiler: This sensor is composed of a mechanically actuated transducer which can be sequentially oriented at different angles and produces a series of range measurements. Usually, the size of the scan sector can be selected from a few degrees to a complete 360° scan around the sensor, which is particularly interesting for obstacle detection tasks. When mounted in a down-looking position, they can also be employed to collect bathymetric data.

Multibeam echo sounder: This sensor is specifically designed to produce bathymetric maps of large areas of the seabed. It is composed of an array of hydrophones which can emit fan shaped beams towards the

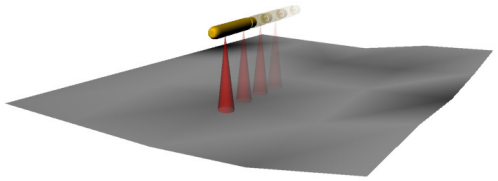
bottom and measure the range of a strip of points placed perpendicularly to the direction of the vehicle's movement. These measurements can be produced at a high rate and resolution.

The sonars in the second category are capable of measuring the returning echo intensity values from particular places within the insonified area. These measurements can then be recomposed into an acoustic representation of the environment generally referred to as an acoustic image. The most common types of imaging sonars are (Figure 2.3):

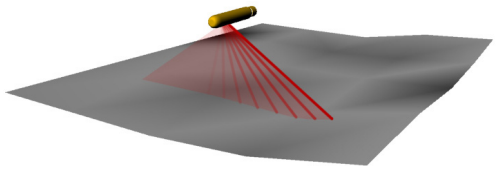
Mechanically scanned imaging sonar: Similar to the mechanically scanning profiler, this device also has an actuated rotatory transducer which can emit fan-shaped beams at different orientations. It is usually placed in a vertical position so it can perform the scanning in the horizontal plane. These devices generally have a configurable scan sector and it is not unusual to find models which can perform full 360° scans, making them perfect for detecting objects around the vehicle. Its main drawback is the slow refresh rate. The operation of this type of sonar has been a major subject during the elaboration of this thesis. Further description of mechanically scanned imaging sonars can be found in Chapter 4.

Electronically scanned imaging sonar: Also known as *multibeam imaging sonar* and *forward-looking imaging sonar*, this sonar is equipped with an array of hydrophones which allows, with the emission of a single pulse, producing the complete acoustic image of the insonified area. This area is usually limited to a small sector in the front of the sensor, but can be scanned at very high rates. Its main drawback is the cost which can be around ten times the price of a mechanically scanned unit.

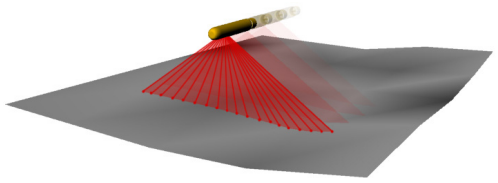
Sidescan sonars: This sonar is designed for imaging large seabed areas. Its mode of operation is analogous to that of multibeam echo sounders, but oriented to imaging tasks. While the sonar is moved along a survey path (either mounted on a vehicle or towed by a ship), it emits fan shaped pulses down toward the seabed across a wide angle perpendicular to the direction of the movement, producing a strip of echo intensity measurements.



(a) Echo sounder.

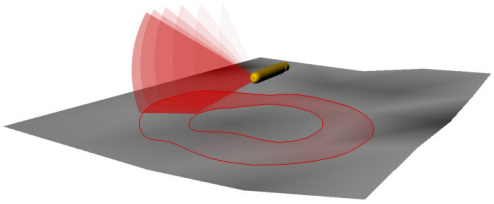


(b) Mechanically scanned profiler.

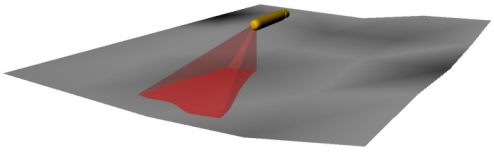


(c) Multibeam echo sounder.

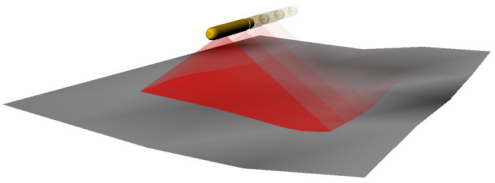
Figure 2.2: Range sonar typology.



(a) Mechanically scanned imaging sonar.



(b) Electronically scanned imaging sonar.



(c) Sidescan sonar.

Figure 2.3: Image sonar typology.

2.4 Underwater SLAM

This section presents an overview of the most relevant contributions to the application of SLAM techniques in underwater environments. This overview does not seek to be an exhaustive enumeration of all the publications in the field but will serve to identify the actors, their approaches and, particularly, those projects involving the implementation of SLAM systems in realistic operating conditions. Later in this chapter, Table 2.1 provides a summary of the different aspects of the works to be commented.

The first implementations of SLAM frameworks using real sensor data can be dated to the later 1990s. In September of 1997, a test rig mounted on the side of a converted U.S. navy freighter was used to acquire a data set as part of a collaborative project between the Naval Undersea Warfare Center (NUWC) and the Groupe d'Etudes Sous-Marines de l'Atlantique (GESMA). The test rig was equipped with a complete sensor suite which included a custom developed sonar, the High Resolution Array (HRA) forward looking imaging sonar, as well as other typical AUV systems such as an INS, a DVL,

a Differential GPS (DGPS) receiver and a sidescan sonar. The acquired data set was later used to test two SLAM approaches. The first one [Carpenter, 1998], presented a simplified EKF implementation in which independent filters were initialized with every new landmark extracted from the HRA acoustic images. The state vector of each of these filters contains the estimate of the vehicle and the corresponding landmark. Although an improved vehicle estimate can be periodically obtained by fusing the estimates from all the independent filters, the landmark estimates remain decoupled and hence, the correlations are ignored. A key aspect of this work is the procedure to obtain the landmarks. First, the acoustic image is segmented to extract the different objects. Then, a set of characteristics (perimeter, area, area-to-perimeter ratio and radial signature) are obtained for each object to produce a description that, in conjunction with a pre-defined similarity metric, makes the data association process possible. In the results section, the estimated trajectories were represented together with the DGPS measurements as ground truth. There were, however, no uncertainty bounds represented. The second work using this same data set was made in collaboration with MIT researchers [Leonard et al., 2001]. They presented an EKF based framework which produced a full stochastic map and hence, correlations were taken into account. In this approach, the measurements from the dead-reckoning sensors were included in the process to improve the quality of the estimation. Moreover, point landmarks corresponding to the centroids of the objects were extracted from the acoustic data and used to build the map. The data association process was carried out by applying a Nearest Neighbor (NN) gating among compatible landmarks. Unlike the previously presented work, the error plots were represented within uncertainty bounds, demonstrating the correct operation of the system. Among these introductory works, it is also worth mentioning [Leonard and Feder, 2001], where the Decoupled Stochastic Mapping (DSM), a computationally efficient approximation to large-scale SLAM, was presented. This proposal reduces the computational burden by dividing the environment into multiple overlapping submap regions, each with its own stochastic map. To assess the performance of the approach, experiments were carried out in simulation, but also with real data obtained in a water tank using a mechanically scanned sonar mounted on a robotic positioning system.

During the same period in which these works took place, the first steps towards underwater SLAM were also taken at the University of Sydney. This time, the chosen test platform was Oberon, a small research underwater vehicle equipped with a Tritech SeaKing mechanically scanned imaging sonar as its principal sensor for the perception of map features. Initial experiments took place in a swimming pool with many artificial landmarks placed

at known positions [Newman, 1999]. Again, point features were selected to represent the observation of these objects in the resulting map. This setup provided the means to test a SLAM framework called the Geometrical Projection Filter or GPF, an approach which estimates the relationships between individual landmarks rather than estimate the location of landmarks in global coordinates. Later, new experiments took place in a real natural terrain along the coast of Sydney, Australia [Williams et al., 2001].This time, a classical EKF implementation of the stochastic map was the core of the SLAM system. Once more, artificial landmarks were deployed in the area to produce a set of reliable point features and although a few natural landmarks were detected, many of them were found too unstable to be incorporated into the map.

A third focus of research on underwater SLAM appeared soon after at Heriot-Watt University [Tena et al., 2001; Tena, 2001]. Following the path opened by [Carpenter, 1998], part of their efforts revolved around the study and development of techniques to characterize the landmarks from acoustic images. In addition to the point coordinates, a vector of landmark characteristics (including size, perimeter, compactness, maximum dimension, centroid and invariant moments) was introduced to improve the data association process [Tena, 2001]. To validate the proposal, experiments were carried out in three distinct environments. First, a data set was obtained in a water tank at the Ocean System Laboratory with a Tritech SeaKing sonar mounted on a planar Cartesian robot using two cylinders as targets. Second, a Seabat 6012 multibeam imaging sonar carried by two divers was operated among the legs of a pier at the Northern Lighthouse Board in Oban, Scotland. And third, a data set was acquired during an open sea trial with a concept electronically scanned sonar mounted on the Ocean Explorer AUV from the Florida Atlantic University. These experiments demonstrated that landmark descriptors are specially useful with real non-artificial data, but that they are much less reliable in situations where landmarks have similar descriptors (e.g. the pier legs). Its worth noting that in this work, SLAM was performed using only an appropriate vehicle model and the sonar data without control inputs or dead-reckoning information.

During the following years, works of the above mentioned groups were alternated with contributions from new researchers. In [Newman and Leonard, 2003], a SLAM framework was proposed to simplify the operation of LBLs by making on-the-fly calibration of submerged transponders using range-only measurements possible. The input of the system is the time of flight between a transceiver mounted on the vehicle and a transponder lying on the seabed whose location is undetermined. Assuming that the altitude of the vehicle is known, the range measurement constrains the possible transponder loca-

tion to a 2D circle on the seabed. When measurements are obtained from different vehicle positions, the intersection of their circles makes determining the position of the transponder possible. Of course, solving this problem involves finding the location of many transponders as well as a sequence of position states. This is done by using a non-linear least-squares optimization algorithm. The data set used to test the approach was produced during the 2002 GOATS experiments, a collaboration between MIT and the NATO SACLANT Underwater Research Center which took place near the cost of Italy. A network of four small LBL transponders were deployed and surveyed with a high precision DGPS system to establish the ground truth. Then, the Caribou AUV, an Odyssey III class vehicle equipped with a transceiver, was operated within the area using an EKF based system for navigation. The EKF integrated compass, Doppler velocity and LBL data with *a priori* knowledge of the transponder locations. To validate the proposal, a comparison between the EKF estimated trajectory and the one obtained with the range-only SLAM is shown. Moreover, the estimated transponder positions are placed within reasonable error bounds when compared with the surveyed ones. A different approach to solve the range-only problem is taken in [Olson et al., 2004], where the authors, using the same 2002 GOATS data set, present an algorithm that imposes geometric constraints on the acoustic measurements to reject outliers. Then, a voting scheme implemented with a two-dimensional accumulator similar to that used in a Hough transform [Illingworth and Kittler, 1988], is in charge of estimating the initial LBL beacon positions. Once their approximate location is obtained, a conventional EKF refines the estimates of both vehicle and beacon positions as new measurements arrive. When represented against the ground truth, the final estimation of the vehicle's path and the position of the transponders show a significant improvement with respect to the results presented in [Newman and Leonard, 2003]. The data collected during the 2002 GOATS experiment is used again in [Newman et al., 2003]. This time, the MIT Synthetic Aperture Sonar (SAS) carried in the nose of the Caribou AUV during the trials, acts as the primary sensor for landmark detection. The paper presents an implementation of the method previously described in [Leonard and Newman, 2003], the Constant Time SLAM (CTS). The CTS algorithm is a consistent and convergent method for updating and creating a set of local maps while determining their best global location estimate. This offers computation independent of the workspace size at the cost of producing a suboptimal solution. The detection and tracking of features from the SAS data was performed automatically using the technique described in [Rikoski and Leonard, 2003]. Data association was performed manually. To evaluate the method, the estimated trajectory resulting from the CTS algorithm is compared with

the solution from a full covariance SLAM along with the ground truth from the AUV's navigation system.

In another line of research, imagery from sidescan sonar has been used to deal with the underwater SLAM problem [Tena et al., 2003a]. Again, the data set employed in this work is the result of a GOATS campaign, in particular from its year 2000 edition. The method relies on a classical EKF implementation of the stochastic map whose estimated trajectory is then smoothed with a Rauch-Tung-Striebel (RTS) filter. In [Tena et al., 2004], the same approach is tested in simulation and with real data obtained during the BP 2002 experiments carried out by the SACLANT in La Spezia, Italy, with a Remus AUV. In both works, the stochastic map stores the location of landmarks extracted from the sidescan sonar images. This landmark extraction is performed manually along with the data association process. In [Tena et al., 2003b], the work is extended by addressing the problem of automatic extraction and association of landmark observations. The automatic extraction technique uses a Markov Random Field (MRF) model to detect candidates [Reed et al., 2004]. After removing those candidates with dimensions outside an acceptable range, a Co-operating Statistical Snake (CSS) is employed to extract the object highlighting and shadow regions. For the data association task, the Joint Compatibility (JC) test [Neira and Tardós, 2001] is improved using the height of the landmark as an additional descriptor. Since no ground truth is available for validation, the performance of the proposal can only be appreciated by comparing the sidescan mosaics created using the navigation data with those generated using the estimated trajectory from the SLAM system. In the different tests, the resulting mosaic from the SLAM system generally offers a higher visual quality and the trajectories are smoother.

The estimation of the vehicle's motion using image mosaicking techniques has many points in common with the problem of underwater SLAM [Negahdaripour and Xun, 2002; García et al., 2003; Gracias et al., 2003]. Among the first approaches which applied such techniques in the form of a SLAM system we can find work done here by the Computer Vision and Robotics Group at the University of Girona. In [García et al., 2002], the authors demonstrated in simulation, a pose-based SLAM composed of a mosaicking system and an augmented-state Kalman filter with the ability to detect crossovers and hence, to perform loop closures. The number of published works presenting underwater visual SLAM systems with real data is still reduced. In 2004, researchers from the University of Sydney presented a SLAM system capable of fusing information from sonar and vision systems to provide estimates of the vehicle's motion and generate image mosaics containing a gross tridimensional model of the scenario [Williams and Mahon, 2004]. The data set for

this work was acquired with the Oberon vehicle during a trial on the Great Barrier Reef in Australia. A camera mounted in a down-looking position was able to capture clear images of the coral reefs over which the vehicle travelled. Simultaneously, a pencil beam scanning sonar mounted above the camera produced a set of range measurements which were used to generate profiles of the terrain directly below the vehicle. The landmarks to be initialized in the stochastic map were extracted from the sonar measurements. For each range measurement, a high contrast visual feature was identified in the image within the area insonified by the sonar. The 3D position of the feature was then incorporated in a EKF framework and tracked using the Lucas and Kanade feature tracking technique [Lucas and Kanade, 1981] to provide the SLAM algorithm bearing only observations of the feature positions. Again, no positioning systems were available for ground truth validation. Therefore, the performance of the system could only be examined by studying the fidelity of the tridimensional mosaic. Despite the reduced dimensions of the resulting terrain model, a significant correspondence between the bumps in the model and the coral structures could be appreciated. This same year, a second proposal for visual underwater SLAM was presented as the result of a joint work between researchers at MIT and the Woods Hole Oceanographic Institution [Eustice et al., 2004]. This proposal consisted of an augmented state EKF which stores the history of the vehicle's positions where a set of camera images were obtained. Then, pairwise image based registration was carried out to determine the correspondences between consecutive images and hence, provided relative measurements between positions. In addition, the measurements from other sensors (heading, depth, velocity, etc.) were also incorporated as observations of the current state of the vehicle. It is worth noting that the system is only capable of detecting correspondences between consecutive images and that correspondences between cross-track images are established manually. The experimental results endorsing the proposal were produced from a data set collected at the Stellwagen Bank National Marine Sanctuary by the SeaBED scientific AUV. Although the experiment was much longer, only a sequence of 100 images taken along a 100 meter trail was used. The resulting vehicle trajectory was compared with the one obtained with dead-reckoning. As expected, the error increased at a lower rate in the SLAM solution, being bounded in those places were cross-track image links happen. Soon after, in 2005, the same research group presented an evolution of the algorithm for visual underwater SLAM [Eustice et al., 2005]. A data set obtained during a survey of the RMS Titanic wreck by the Hercules deep-sea ROV served as the testbed for a SLAM framework based on the use of the EIF instead of the traditional EKF approach. The sparsity of the solutions from an EIF led to computationally efficient SLAM

algorithms and made storing a larger number of elements in the state possible. This is demonstrated in the work at hand with a state vector containing 866 robot states, each one corresponding to the acquisition of one of the images from the data-set. A second enhancement of the proposal is that the search for correspondences is not restricted to consecutive images anymore, since the system is capable of determining the regions where correspondences with other images can occur.

This same year, [Roman and Singh, 2005] proposed a method to improve bathymetric mapping using SLAM. The solution consisted of generating a set of submaps from small bathymetric patches created over short periods of time. In a similar way to that of the presented visual underwater SLAM algorithms, estimates of previously visited vehicle positions are retained in the state of an EKF to serve as anchor points for these submaps. Then, these patches are registered to generate relative position measurements between delayed states. Its worth mentioning that during the creation of the submaps, an accumulation of navigation errors occurs which affects the position of the range measurements. To minimize their effect, the authors propose a method to identify the quantity of internal errors in the submap to determine the point in which to break the current map and start a new one. A similar criteria is applied to ensure that the submap contains enough 3D information to make the registration process possible. In the experimental part of the work, the Jason ROV, equipped with an SM2000 multibeam echo sounder from Kongsberg-Mesotech Ltd. and many other typical navigation sensors, performed a 12 hour survey over a hydrothermal vent site. The resulting data set was used to generate a bathymetric map of the zone using both dead-reckoning and SLAM estimated trajectories. Using measurements from an LBL system as ground truth, it is possible to observe a better alignement of the SLAM estimated positions than those estimated by dead-reckoning. Moreover, the bathymetric map resulting from SLAM presents significantly fewer registration errors and higher detail.

In 2006, a SLAM system running on the Tri-Dog 1 AUV was successfully deployed during sea experiments carried out at the Tagari vent area of Kagoshima Bay in Japan [Maki et al., 2006b]. The system took advantage of bubble plumes present in the area as well as two sonar reflectors specifically deployed to serve as landmarks. A mechanically scanned profiling sonar was the primary sensor for landmark detection. The sonar was set to scan horizontally around the vehicle to map the positions of the plumes and reflectors in the form of point landmarks. Moreover, other dead-reckoning sensors were also used. The SLAM framework, whose description and preliminary water tank tests were presented in [Maki et al., 2006], runs a particle filter which estimates the vehicle state while, simultaneously, a map builder is in charge

of incorporating newly detected landmarks into the map. A third element, the motion controller, generates the control signals necessary to drive the vehicle through a sequence of waypoints which define the autonomous mission. Although the system effectively builds a map and simultaneously uses its information to localize the AUV, it is not able to update the map nor does it take into account the uncertainty of the vehicle's position while initializing the features in the map. The experimental part of the work presents several trajectories obtained during various dives as well as an example of an image mosaic composed using only the SLAM estimated positions. There is however no ground truth to validate the results.

During the last year, one more approach on underwater SLAM has appeared. [Fairfield et al., 2007] presents the DepthX, an AUV especially designed for exploring *cenotes* (i.e. sinkholes) in Mexico. Equipped with an array of 56 narrow beam sonar transducers, the vehicle is capable of acquiring a constellation of range measurements all around it and therefore, perceive tridimensional structures from the environment. In additon, dead-reckoning sensors assist the navigation. In this work, the core of the SLAM system is a Rao-Blackwellized particle filter in which each particle represents the vehicle's position and the map [Grisetti et al., 2007]. The map is stored within a 3D evidence grid which uses the Deferred Reference Counting Octree (DCRO) data structure to reduce the memory requirements [Fairfield et al., 2007b]. This paper presents two experiments in which the SLAM system has been tested. The first one was performed in a large cylindrical water tank (11.6 meters deep and 16.8 meters in diameter) using the dead-reckoning sensors for real time navigation. Then, the same data-set was used in a localization algorithm together with a previous map of the tank to produce a ground truth trajectory. This trajectory serves to validate the SLAM system when, finally, an offline version is executed. The second experiment consisted of a dive performed in the La Pilita cenote with a real time version of the SLAM algorithm running on the DepthX vehicle. Due to the lack of ground truth, it was not possible to make strong assertions about the accuracy of the system. However, the vehicle succeed in creating a map of the cenote and its navigation was accurate enough to return the vehicle to the starting location after the test.

2.5 Discussion

This chapter has introduced the SLAM problem, its history and, in particular, a selection of the most representative works carried out so far towards a

Table 2.1: Summary of selected works on underwater SLAM

	Principal sensor	Landmarks and scenario	Filtering	Map	Feature extraction	Data association	Ground truth
[Carpenter, 1998]	Imaging sonar (electr.)	Natural (sea)	EKF (decoupled)	Landmark (point)	Automatic	Automatic	Yes
[Newman, 1999]	Imaging sonar (mech.)	Artificial (pool)	EKF (GPF)	Landmark (point)	Automatic	Automatic	Yes
[Tena et al., 2001]	Imaging sonar (mech.) Imaging sonar (electr.)	Artificial (tank) Artificial (pier)	EKF	Landmark (point)	Automatic	Automatic	Yes No
[Tena, 2001]	Imaging sonar (mech.) Imaging sonar (electr.) Imaging sonar (electr.)	Artificial (tank) Artificial (pier) Natural (sea)	EKF	Landmark (point)	Automatic	Automatic	Yes No No
[Leonard et al., 2001]	Imaging sonar (mech.) Imaging sonar (electr.)	Artificial (tank) Natural (sea)	EKF	Landmark (point)	Automatic	Automatic	Yes Yes
[Williams et al., 2001]	Imaging sonar (mech.) Imaging sonar (mech.)	Artificial (pool) Mixed (reef)	EKF	Landmark (point)	Automatic	Automatic	Yes No
[Leonard and Feder, 2001]	Imaging sonar (mech.)	Artificial (tank)	EKF (DSM)	Landmark (point)	Automatic	Automatic	Yes
[Tena et al., 2003b]	Sidescan sonar	Natural (sea)	EKF	Landmark (point)	Automatic	Automatic	No
[Newman and Leonard, 2003]	LBL (range only)	Artificial (sea)	LS optimization	Landmark (point)	Automatic	Solved	Yes
[Tena et al., 2003a]	Sidescan sonar	Natural (sea)	EKF/RTS	Landmark (point)	Manual	Manual	No
[Newman et al., 2003]	Imaging sonar (SAS)	Natural (sea)	Undefined (CTS)	Landmark (point)	Automatic	Manual	Yes
[Olson et al., 2004]	LBL (range only)	Artificial (sea)	EKF	Landmark (point)	Automatic	Solved	Yes
[Tena et al., 2004]	Sidescan sonar	Natural (sea)	EKF/RTS	Landmark (point)	Manual	Manual	No
[Eustice et al., 2004]	Camera	Natural (sea)	EKF	Vehicle poses	Automatic	Mixed	No
[Williams and Mahon, 2004]	Camera and profiler	Natural (reef)	EKF	Landmark (point)	Automatic	Automatic	No
[Eustice et al., 2005]	Camera	Artificial (wreck)	IF	Vehicle poses	Automatic	Automatic	No
[Roman and Singh, 2005]	Multibeam echo sounder	Natural (sea)	EKF	Bathymetric sub-maps	–	Automatic	Yes
[Maki et al., 2006b]	Profiler (mech.)	Mixed (sea)	PF	Landmark (point)	Automatic	Automatic	No
[Fairfield et al., 2007]	56 pencil beams	Artificial (tank) Natural (cenote)	PF	Evidence grid	–	Automatic	Yes No

solution for underwater environments. Autonomous navigation of underwater vehicles has been a subject of great interest for years with an abundant published bibliography on many different topics, from dead-reckoning to terrain aided navigation. However, the underwater SLAM is still in its initial phase and a relatively reduced number of approaches have been presented. As has been shown in Section 2.4, the majority of underwater SLAM approaches have some points in common. For instance, among the different sensorial options, imaging sonars seem to be the most common choice, probably because of their high capability to explore large areas in search of features. Although cameras can produce rich information, their use is restricted to a more local domain and, in some situations, they may suffer visibility problems. Profilers and range sensors in general, are less suitable for feature extraction and using their raw data (e.g. for scan matching) usually requires accumulating measurements in order to produce a sufficient representation of the scenario. The use of imaging sonars seems to be evenly distributed between those scanned mechanically and those scanned electronically. However, a direct relationship between electronically scanned sonars and those SLAM examples performed with AUVs in real environments can be observed . On the other hand, mechanically scanned devices are usually employed during tests performed under lab conditions. Independently of cost considerations, this may be related to the different scanning rates of the devices. Electronically scanned sonars can produce images almost instantaneously and the distortions due to the vehicle's motion usually fall within the range resolution of the sonar and so their effect can be ignored. The use of mechanically scanned sonars, with a much slower scanning rate, is usually limited to platforms which are static or moving at low velocities or in those situations where a suitable dynamics model and sufficient dead-reckoning sensors are available.

Another remarkable aspect of the studied SLAM approaches is the predominance of the Kalman filters over other estimation techniques such as information filters or particle filters. Although the latter have gained importance during recent years, the number of implementations is still reduced. Moreover, the major part of the proposed systems implement variants of the classical stochastic map in which point features are used as landmarks. Those features generally correspond to the centroids of objects which appear as high echo-intensity zones in the acoustic images. Extracting these kind of features from the seabed is difficult for many reasons. For instance, they have variable sizes and their shapes can change depending on the sensor's vantage point. Of course, this makes the landmarks less reliable and could induce errors in the estimation of their centroids. Another typical problem of the underwater SLAM is the landmark shortage. The sea bottom is generally flat and it can be quite difficult to find landmarks capable of producing dis-

tinguishable features in the images. For this reason, it is not uncommon to find applications in which reliable artificial features are deployed to improve the performance of the system. Among alternative map representations, we can find maps composed of camera images. These maps contain rich visual data and in some cases, tridimensional information is also incorporated. Although at this point there are still few published works, the interest in image mosaicking techniques and their possible applications will probably foster future works. On the other hand, two approaches using dense range data have also been presented. These systems produce very detailed maps, however, their main drawback is that their use remains restricted to areas where it is possible to extract sufficient 3D information to ensure the success of the registration process.

With respect to the validation of the SLAM proposals, it is worth mentioning that in half of the reported SLAM examples ground truth is not available, while an important part of the remaining half corresponds to those tests carried out in lab conditions. Obtaining reliable ground truth in real scenario operations is very complex. It usually requires the deployment and calibration of additional acoustic localization systems such as LBL or USBL. Nevertheless, these systems can only provide ground truth for the vehicle trajectory and validating mapping results is even more difficult, since previous maps of the area rarely exist.

2.6 Aims of this work

As pointed out in the previous chapter, the work done throughout this thesis pursued the goal of developing a SLAM system for AUVs operating in man-made environments. Among the sensorial options available, the mechanical scanned imaging sonar has been chosen because of its versatility but also for its relative low cost, in particular, when compared with the electronically scanned ones. This approach presents both advantages and disadvantages. For example, navigating through this kind of environment guarantees the presence of underwater structures from which is possible to extract reliable features. An advantage of mechanical sonars in contrast with those electronically scanned is that their perception area is not limited to the front of the vehicle. In fact, they can continuously rotate 360° around the vehicle, which is perfect for those situations where a limited number of landmarks exist. On the other hand, using a mechanically scanned sonar mounted on a moving vehicle requires dealing with motion induced distortions in the acoustic data. The effect is particularly problematical because those structures found in the application at hand are generally walls which appear as linear shapes

in the acoustic images. In the presence of distortions, lines can experience important deformations that may preclude their correct identification, contrary to what will happen when working with point landmarks, which will be, at most, incorrectly located.

Chapter 3

Design and development of the Ictineu AUV

This chapter describes the Ictineu AUV (Figure 3.1), the research vehicle of the Computer Vision and Robotics Research group at the University of Girona that constitutes the experimental platform of this thesis.

In 2006, the Defence Science and Technology Lab (DSTL), the Heriot-Watt University and the National Oceanographic Centre of Southampton organized the first Student Autonomous Underwater Challenge - Europe (SAUC-E) [DSTL, 2006], an European-wide competition for students to foster research and development in underwater technology. Ictineu AUV was originally conceived as an entry for the SAUC-E competition by a team of students collaborating with the Underwater Robotics Laboratory [VICOROB, 2006; Ribas et al., 2007]. This author, who was team leader during the competition, got involved in the hardware design and construction phase as well as in the development of its sonar based localization system (described in Section 5.2). Although the competition determined many of the vehicle's specifications, Ictineu was also designed keeping its posterior use as an experimental platform for various research projects in our laboratory in mind. The experience obtained from the development of previous vehicles in the group, Garbi ROV [Amat et al., 1998], Uris [Batlle et al., 2004] and Garbi AUV, made it possible to build a low-cost vehicle of reduced weight (52 Kg) and dimensions (74 x 46.5 x 52.4 cm) with remarkable sensorial capabilities and easy maintenance.

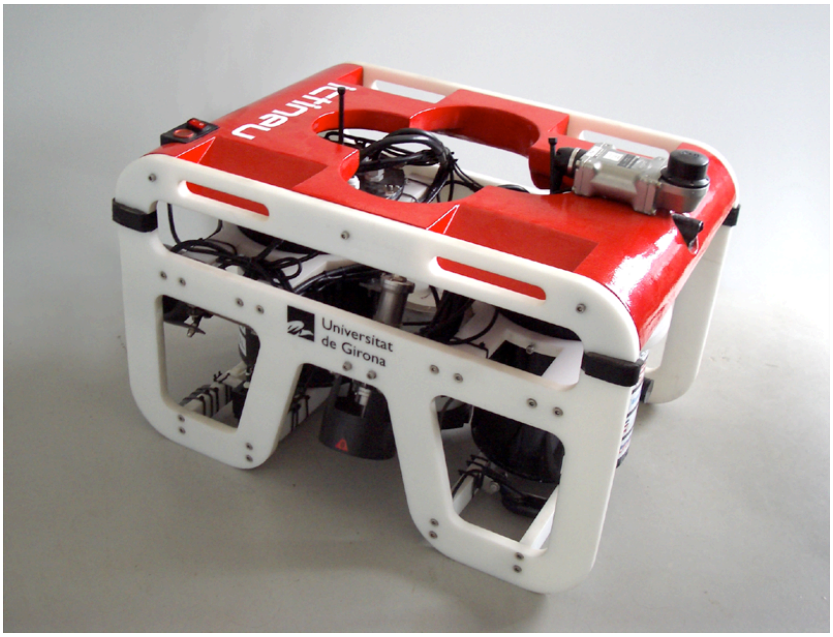
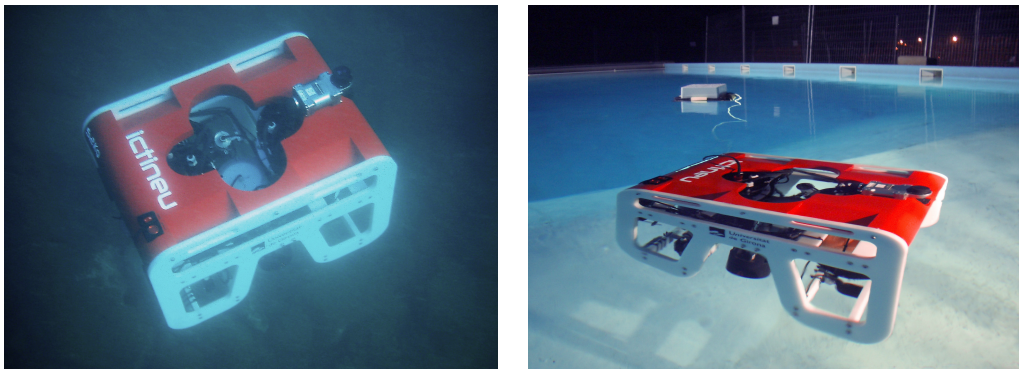


Figure 3.1: The Ictineu AUV, a research vehicle of the Underwater Robotics Laboratory of the University of Girona.

3.1 Mechanical Aspects

The Ictineu AUV was conceived around a typical open frame design. This configuration has been widely adopted by commercial ROVs because of its simplicity, robustness and reduced cost. Although the hydrodynamics of open frame vehicles is known to be less efficient than that of closed hull type vehicles, they are suitable for applications not requiring movements at high velocities or traveling long distances. The robot chassis is made of Delrin, an engineering plastic material which is lightweight, durable and resistant to liquids. Another aspect of the design is the modular conception of its components which simplifies upgrading the vehicle and makes it easier to carry out maintenance tasks. Some of the modules (the thrusters and the major part of the sensors) are watertight and therefore, are mounted directly onto the vehicle chassis. On the other hand, two cylindric pressure vessels made of aluminum house the power and computer modules while a smaller one made of Delrin contains the Motion Reference Unit (MRU). Their end caps are sealed with conventional O-ring closures while the electrical connections with other hulls or external sensors are made with plastic cable glands sealed with epoxy resin. The Ictineu is propelled by four thrusters. Two of them,



(a) Untethered mode. (b) Equipped with a buoy.

Figure 3.2: Different configurations of the Ictineu AUV.

mounted horizontally, can propel the vehicle in the surge Degree Of Freedom (DOF), as well as change the heading (yaw DOF). The two vertical thrusters are disposed in a particular slanted distribution which makes it possible not only to control the movement in the heave DOF but also to produce small lateral movements in the sway DOF. Therefore, the prototype is a fully actuated vehicle in four DOF (surge, sway, heave and yaw), while being passively stable in roll and pitch as its meta-centre is above the centre of gravity. This stability is the result of an accurate distribution of the heavier elements at the lower part of the chassis combined with the effect of technical foam placed in the top, that with its 10.5 liters volume and a weight of 0.6 Kg provides a slightly positive buoyancy to the Ictineu.

3.2 Tethered/Untethered Modes of Operation

The Ictineu can operate either as a ROV (tethered mode) or as an AUV (untethered mode). An optional 30 meter umbilical cable can be connected to the two principal hulls to supply power and communications to the vehicle. This mode of operation is very useful not only to operate the Ictineu as a ROV but also while working under laboratory conditions as it makes testing/developing the vehicle's software over long time periods possible. An external 1200 VA power supply, derived from 230 V AC mains or a power generator, feeds about 24 V DC to the vehicle. Moreover, a standard ethernet connection is embedded in the umbilical allowing direct data transmission with the onboard computers. This 100 Mbps connection makes it possible not only to remotely operate the vehicle but also to transmit real time video

streams from the vehicle's cameras. When working in full AUV mode, the umbilical cable is removed and the connectors in the hulls are sealed with plugs. Being untethered, the vehicle relies on batteries to power all the systems and therefore, has a limited running time but a longer range of operation. Unfortunately, communications with the vehicle are not possible in this setup. Recently, a third mode of operation has been developed. A buoy, connected to the Ictineu through a 25 meter ethernet cable and equipped with a 802.11g Wi-Fi access point, enables communication with the vehicle while still providing a high level of autonomy. In addition, the buoy is also equipped with its own batteries for power supply and a DGPS receiver which provides the position of the buoy and hence, the approximate location of the vehicle in the global reference frame.

3.3 Power Module

The power module (see Figure 3.3) contains the four power drivers for the thrusters as well as a pack of 2 cheap, sealed 12V 12Ah lead acid batteries that can provide the Ictineu with over 1 hour of running time. A DC-DC converter is included to provide stabilized voltage to the rest of the components. Finally, a simple relay circuit commutes between the internal and the external power supplies when the umbilical cable is connected. If necessary, it is also possible to recharge the batteries with the external power from the umbilical cable.

3.4 Computer Module

Two PCs, one for control and one for image and sonar processing connected through a 100 MBs switch, form the core of the robot's hardware. An image of the complete power module can be seen in Figure 3.4. The control PC is an AMD GEODE- 300MHz powered by a 50 W power supply module. The PC104 stack also incorporates an A/D and digital I/O card with 8 analogue input channels, 4 analogue output channels and 24 digital I/O. The mini-ITX computer is a Via C3 1 GHz Pentium clone and is used to process the data from the imaging sonar and the cameras. A cheap PCTV110 from Pinnacle is used for image processing.

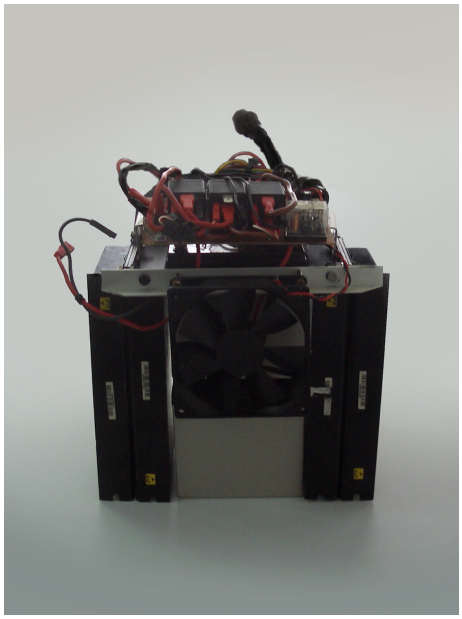


Figure 3.3: Power module

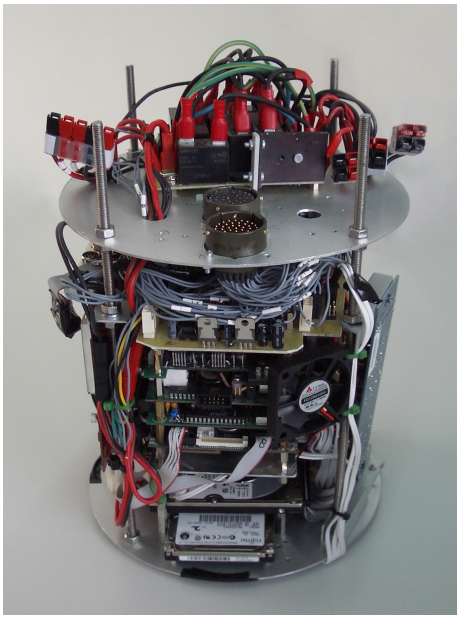


Figure 3.4: Computer module

3.5 Actuators

Since the Ictineu AUV is not an intervention robot, the actuator suite is basically limited to the 4 thrusters necessary to operate the vehicle. However, with the objective of taking part in the SAUC-E competition, an additional actuator, a marker dropper, was incorporated.

3.5.1 Thrusters

Each one of the thrusters mounted in the Ictineu vehicle (Figure 3.6) is able to produce around 14.7/14.2 N of forward/backward thrust. They were built with 250 W Maxon DC motors equipped with planetary gears and are enclosed in stainless steel housings with O-ring sealed end caps. The rotation is transmitted to a three-bladed brass propeller by means of a stainless steel shaft mounted through a rubber lip seal. Although lip seals do not withstand high pressures, their simplicity and reduced cost make them a good choice for vehicles operating in shallow waters.

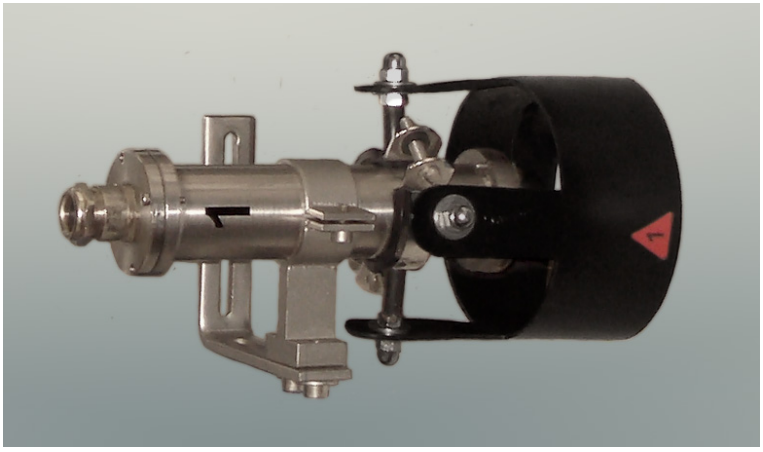


Figure 3.5: One thruster from the Ictineu.

3.5.2 Marker dropper

One of the missions proposed for the SAUC-E competition consisted of locating a circular target lying on the bottom of the tank and hitting it with a marker dropped from the vehicle. An electromagnetically actuated release mechanism with a 3-shot magazine was designed for this purpose and mounted in the rear part of the vehicle.

3.6 Sensor Suite

One of the main objectives the team had in mind while designing the Ictineu was to provide the vehicle with a complete sensor suite. Taking the set-up in the Garbi AUV as starting point, the new suite was created by adding new sensors to correct some limitations of the old prototype. Moreover, as the Ictineu was started from scratch, it was possible to improve the distribution of the acoustic sensors within the vehicle frame in order to avoid dead zones and improve their overall performance.

3.6.1 Miniking Imaging Sonar

The Tritech MiniKing is a small compact mechanically scanned imaging sonar (MSIS) designed for use in underwater applications such as obstacle avoidance and target recognition for both AUVs and ROVs. This sonar can perform scans in a 2D plane by rotating a fan-shaped sonar beam through a series of small angle steps. It can be programmed to cover variable length

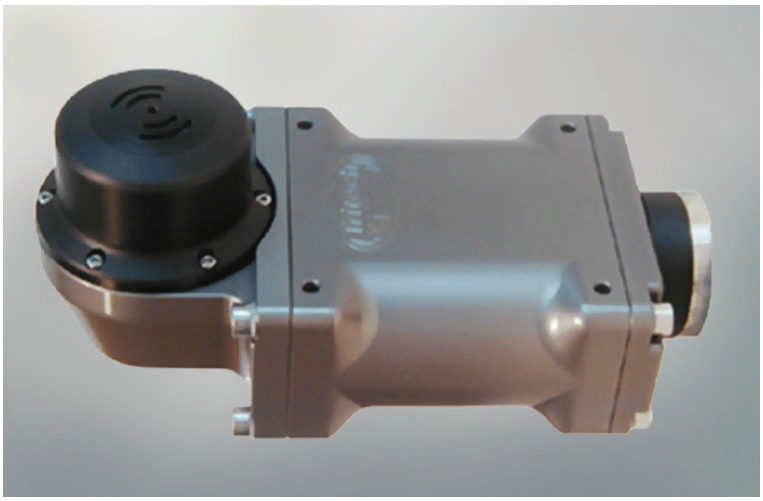


Figure 3.6: The Tritech Miniking imaging sonar.

sectors from a few degrees to full 360° scans. A characteristic fan-shaped beam with a vertical aperture angle of 40° and a narrow horizontal aperture of 3°, allows forming a sonar image with enough information about the surrounding environment to recognize sizes, shapes and surface reflecting characteristics of a target at distances of up to 100 meters. The sensor is mounted on the upper front part of the Ictineu AUV to provide a clear view and avoid occlusions in the resulting data. Its capacity to sense the environment in which the vehicle is operating makes the Miniking one of the most important sensors aboard the Ictineu and presents a valuable potential for underwater localization and SLAM. This has made the Miniking one of the principal objects of study throughout the elaboration of this thesis. A more detailed description of the operation of MSISs in general and of the Miniking sensor in particular can be found in Chapter 4.

3.6.2 Doppler Velocity Log

The SonTek Argonaut DVL is a sensor specially designed for ROV/AUV applications which measures ocean currents, vehicle speed over ground, and altimetry using a precise 3-axis measurement system based on the Doppler shift effect (see Figure 3.7). Moreover, it has the capacity to analyze the quality of the measurements and produce a status value, which makes discarding erroneous data possible. This system operates at a frequency of 1500 kHz and has a range of about 15 m. Its three acoustic transducers are slanted

25° off the housing vertical axis and are equally spaced at 120° relative azimuth angles. In our particular configuration, the device is also equipped with additional sensors:

Compass: Outputs the sensor heading (angle with respect to the magnetic north).

Tilt sensors: Measures the roll (rotation about the X axis) and pitch (rotation about the Y axis) angles.

Pressure sensor: Provides depth data by means of water column pressure measurements.

Temperature sensor: Provides water temperature for internal sound speed calculations, improving the measurements from the acoustic device.

Although some of these sensors could individually produce measurements at a higher frequency, the resulting Argonaut DVL output rate is of about 1.5 Hz because of the limitations of the acoustic device. However, the inclusion of all this equipment converts the Argonaut DVL into a very versatile sensor which, together with its compact size, low power consumption and depth ratings of about 200 meters, makes it especially suited for underwater vehicle navigation.

Figure 3.7: Sontek Argonaut DVL

Figure 3.8: Xsens MTi MRU

3.6.3 Motion Reference Unit

The Xsens MTi sensor (Figure 3.8) is a gyro-enhanced low cost miniature Motion Reference Unit (MRU) which provides 3D orientation (attitude and heading), 3D rate of turn (rate gyro) as well as 3D acceleration measurements. Although the sensor is able to provide data at a higher rate, the system gathers measurements from the MTi at a rate of 10 Hz. In order to produce drift-free angular measurements, the sensor also measures the directions of gravity and magnetic north. Our particular device configuration has 17 m/s^2 full scale in acceleration measurements, which is far from the smaller accelerations that Ictineu actually experiences. For this reason we do not usually rely on its acceleration estimates. On the other hand, the angular measurements are much more reliable and, as they are outputted at a higher rate than the data from the DVL sensor, they have been chosen as the main source for the vehicle's attitude estimation.

3.6.4 Cameras

Ictineu is equipped with two cameras. The first is a forward-looking colour camera mounted on the front of the vehicle and is intended for target detection and tracking, inspection of underwater structures and to provide visual feedback when operating the vehicle in ROV mode. The second camera is a downward-looking black and white camera placed in the lower part of the vehicle. This camera is mainly used to capture images of the seabed for research on image mosaicking.

3.6.5 Hydrophone

The last task to be completed in the SAUC-E mission consisted of surfacing within a designated zone marked by means of an active acoustic device. In order to detect this acoustic signal, an external hydrophone was mounted on the front of the vehicle while all the signal processing circuitry, which was specifically designed for the task, was mounted inside one of the main pressure vessels.

3.6.6 Safety Sensors

There are several minor sensors whose purpose is to ensure the safety of the vehicle. The greater part of these sensors are mounted inside the different pressure vessels and have the mission of measuring temperature, pressure and to detect water leakage. The activation of any of these sensors implies

that some problem is happening inside the pressure vessels and therefore, an alarm is raised to prevent irreparable damages. There is also an acoustic range finder mounted on the front part of the vehicle. The purpose of this device is to provide a simple way to determine the presence of an obstacle close to the vehicle and thus, avoid collisions without having to analyze complex data from the imaging sonar.

3.7 The O^2CA^2 Software Architecture

The software architecture has the task of guaranteeing the AUV's functionality. The real-time POSIX together with the ACE/TAO CORBA-RT ORB have been extensively used to develop the architecture as a set of distributed objects with soft real time capabilities. These objects are distributed between the two onboard PCs and, when operating in thetered mode, the external PC. The architecture is composed of a base system and a set of objects customized for each particular robot, which makes it possible to share the same software architecture with all the vehicles in the lab. There are classes providing soft real-time capabilities to allow for a periodic execution of tasks such as the controllers or the sensors. Another important part of the base systems are the loggers. A logger system is used to log data from sensors, actuators or any other object component. Loggers do not execute in real time, they are background processes which receive the data from real time objects. Their role consists of packing the data and saving them into files. It is worth noting that, although loggers do not run in real time, the data has a time-stamp corresponding to the gather time. Moreover, all the computers in the network are synchronized by means of the NTP (Network Time Protocol) and hence, all the data coming from different sensors can be time related. The software architecture is divided between three modules as represented in Figure 3.9: Robot interface module, perception module and control module.

3.7.1 Robot Interface Module

This is the only module containing software objects that dialog with the hardware. There are basically two types of objects: sensor objects responsible for reading data from sensors and actuator objects responsible for sending commands to the actuators. Sensor objects for the Ictineu AUV include a DVL, an imaging sonar, a MRU, two cameras, a depth sensor, and an echo sounder. There are also objects for the safety sensors like water leakage detectors and internal temperature and pressure sensors that allow for the

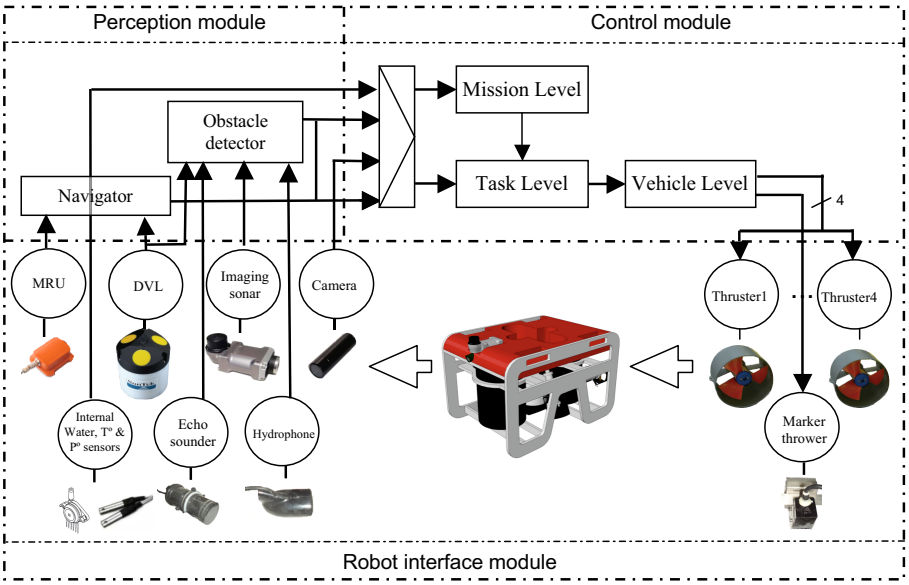


Figure 3.9: Schematic of the Ictineu AUV software architecture.

monitoring of conditions within the pressure vessels. Actuator objects for the Ictineu include the thrusters and the marker thrower.

3.7.2 Perception Module

This module contains two basic components, the *Navigator* and the *Obstacle Detector*. The *Navigator* object has the goal of estimating the position of the robot. To accomplish this task, there is an interface called *Navigation Sensor* from which all the localization sensors (DVL, MRU, depth sensor) inherit. This interface provides all these sensors with a set of methods to return the position, velocity and acceleration in the 6 DOF together with an estimation of the quality of these measurements. The *Navigator* can be dynamically connected to any *Navigation Sensor*, fusing the data to obtain more accurate position, velocity and acceleration estimates. Furthermore, the *Navigator* can also access the imaging sonar to implement the navigation method specifically designed for the SAUC-E competition described in Section 5.2. The control module uses the navigation data provided by the *Navigator* keeping the behaviours independent of the physical sensors being used for the localization. The *Obstacle Detector* uses the same philosophy to provide obstacle position in the world fixed frame. The *Obstacle Detector* is also used to detect the distance between the vehicle and the bottom of the pool. Detecting frontal obstacles is possible using the echo sounder or the

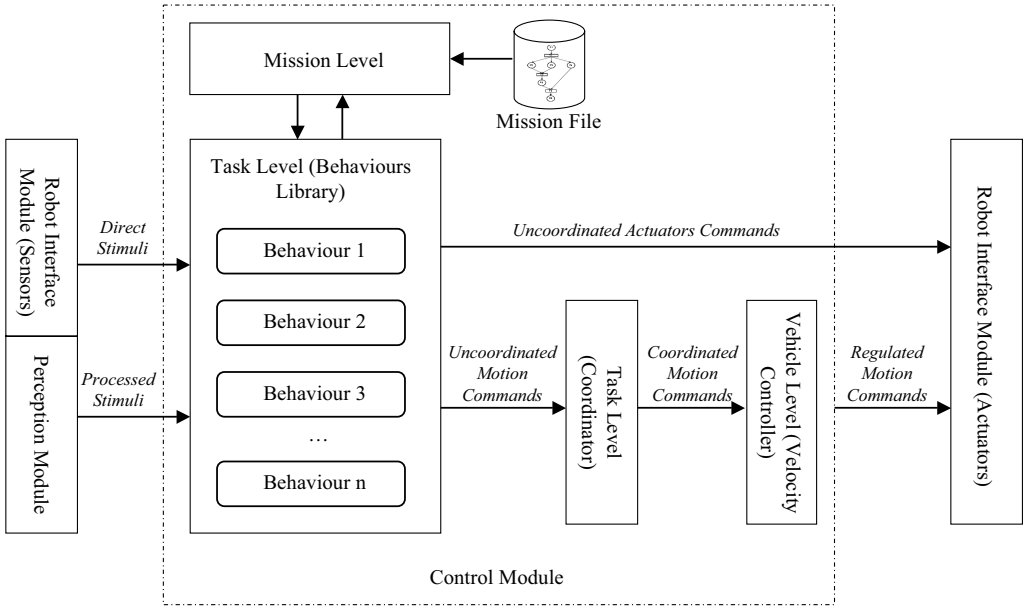


Figure 3.10: Schematic of the Ictineu AUV control architecture.

imaging sonar and the pool bottom obstacles can be detected with the DVL sensor.

3.7.3 Control Module

The control module receives sensor inputs from the perception module and sends command outputs to the actuators residing in the robot's interface module (Figure. 3.10). Since tasks and behaviours are words that are interpreted in different ways for different authors in the literature, hereafter, we describe how they are interpreted within our project:

- A behaviour is a function that maps the sensor input space (stimuli) into a velocity set point (behaviour response) for the robot's low level controller. The behaviour response is chosen in a way that drives the robot towards its corresponding goal. In this way, the goal corresponding to the *KeepDepth* behaviour is considered to be achieved when the robot is within an interval around the desired depth.

- A task is a set of behaviours that are enabled together to achieve a more complex goal. For instance, *KeepDepth* and *MoveTo2D* can work together to allow for planar navigation.

The control module follows the principles of the hybrid control architecture organized in three layers: vehicle level, task level and mission level.

Vehicle Level

The vehicle level is composed of a MIMO PID velocity controller for each DOF. This object reads the vehicle's velocity from the *Navigator* object and receives the velocities setpoints from the *Coordinator* object. This level also includes a simple control allocator strategy based on the pseudo inverse of the thruster configuration matrix [Fossen, 1994].

Task Level

The Task level is a conventional behavioural layer [Arkin, 1998] including a library of behaviours that can run alone or in parallel. Each behaviour has a particular goal. The input of a behaviour can be taken from any object of the software architecture (sensors, perception module...). The output, called behaviour response, contains:

- Velocity setpoints for every DOF normalized between -1 and 1.
- Activation level for every DOF normalized between 0 and 1 indicating how important it is for the behaviour to take control of the robot.
- Blocking term (boolean) stating if the behaviour must block the execution thread of the mission level.

To initialize a behaviour, apart from setting its particular parameters, it is necessary to specify the following attributes:

Enable: Boolean variable that indicates if the behaviour is activated or not and if its output will be considered by the Coordinator.

Priority: Priority stating the relative importance of each behaviour.

TimeOut: The time out indicates when the behaviour will block the execution thread. If TimeOut$<$ 0, the behaviour blocks the execution thread until its goal is fulfilled. If TimeOut$=$ 0, the behaviour doesn't block the execution thread at all. If TimeOut$>$ 0, the behaviour blocks the execution thread until TimeOut seconds elapse or until its goal is fulfilled.

During the execution of a mission, more than one behaviour can be enabled simultaneously. Hence, a coordinator module is used to fuse all the responses corresponding to the enabled behaviours into a single response to be sent to the velocity controller (vehicle level).

Each degree of freedom is considered separately since not all the behaviours act on all the DOF. To combine all the behaviour responses, the *Coordinator* sorts all the responses by their priority combining them two by two for every DOF, from the highest priority to the lowest. To combine the responses, the activation level and a k factor are used as follows:

$$s = \frac{a_1 s_1}{a_1 + a_2(1-a_1)^k} + \frac{a_2 s_2 (1-a_1)^k}{a_1 + a_2(1-a_1)^k}.$$

a_1, a_2 and s_1, s_2 correspond to the activation level and the desired setpoints for the highest priority and the least priority behaviour respectively, while s corresponds to the final coordinator response. The *Coordinator* output, after combining all active behaviours, is a vector as large as the number of the robot's DOFs where each value corresponds to a normalized velocity [Palomeras et al., 2006]. This coordination mechanism can be seen as a hybrid between the classical competition and cooperation methods. When the activation level of the behaviour with highest priority is zero, the coordinated response coincides with the output of the behaviour with lower priority. If the behaviour with the highest priority requests the control of the vehicle using an activation level equal to one, then the response of the behaviour with lower priority (non dominant behaviour) is totally subsumed and the coordinated response matches the one of the behaviour with the highest priority (dominant behaviour). If both behaviours simultaneously request the vehicle control through an activation level greater than zero then both of them are merged using a weighted average operation with the weights depending on the specified activation levels. If equal activation levels are used, the dominant behaviour always has a stronger weight regulated by means of the exponent k. For details about this coordination mechanism the interested reader is forwarded to [Carreras et al., 2001].

Mission Level

Finally, the upper layer (mission level) is responsible for the sequencing of the mission tasks, selecting for each mission phase the set of behaviours that must be enabled as well as their parameters.

The mission controller was built with a Petri network in which the sequence of tasks is defined. Since the vehicle can move in an unstructured environment, unexpected situations have to be taken into account by the

mission designer. According to the network, some nodes will become active. Each node represents a behaviour that will be executed on the task controller. There is a library of behaviours that are used to define a mission. Each one has a simple goal such as move to point, keep depth, search a target, etc. Therefore, the mission controller has the job of defining the task the robot is accomplishing at each moment by activating or deactivating behaviours with the final goal of fulfilling the mission. The mission controller does not determine the actions that guide the robot, it only determines the active behaviours and its configuration which, through the task controller, will be coordinated to guide the robot.

In our Petri net, every place corresponds to one behaviour with a particular configuration. When a place has a token, this behaviour is enabled. When all places that go towards a transition are enabled, and their behaviours do not block the execution thread, the transition is ready to be fired. When a transition is fired, a token is removed from each of the input places of the transition and a token is generated in each output place of the same transition. The control mission algorithm starts on the initial state, checks fired transitions, applies the previously explained procedure, and repeats this process until it reaches the final state [Palomeras et al., 2006].

3.8 Summary and further work

As a research platform, the Ictineu AUV is subject to constant upgrades. These upgrades have the goal of either correcting and improving detected deficiencies or to extending the capabilities of the vehicle. Hereafter, some of the imminent modifications are briefly described.

Thruster upgrade: The thrusters currently mounted in the Ictineu AUV were developed in our lab years ago having a smaller vehicle in mind. Although they are sufficient for operation under lab conditions, missions taking place in natural scenarios may require more thrusting power, specially in the presence of water currents, as well as a better capacity to withstand higher pressures for operations at greater depths. Recently, six new SeaBotix SBT150 thrusters had been acquired. Their reduced dimensions and weight, together with a thrust of about to 22 N make them a good choice for a small vehicle such as the Ictineu. At the time of writing this thesis, four of the thrusters have been mounted horizontally in a slanted distribution which makes performing movements in the surge and sway DOFs possible. Moreover, this particular distribution also provides redundancy to the system and therefore, the vehicle is now able to operate even with a damaged thruster. The

two remaining thrusters have been mounted vertically and actuate the heave DOF.

Absolute positioning: Recently, a Ultra Short Baseline (USBL) has been acquired by the lab. An USBL is a method of underwater acoustic positioning. This device consists of a transceiver, which is usually placed on the surface, on a pole under the ship, and a transponder/responder mounted on the AUV (see Figure 1.1). The device determines the position of the vehicle by calculating the range and angles obtained after the transmission and reply of an acoustic pulse between the transceiver and the transponder/responder. Our particular model can also provide communications by means of an acoustic modem incorporated in the same package. Having a USBL opens the door to many applications where the availability of robot positioning is crucial. Moreover, the output of this device is precise enough (0.2 meter for the range and 0.25 degree for the angle) to be used as the ground truth to test SLAM algorithms. At the present time, software drivers have been developed and integrated into the vehicle's architecture, and a navigation filter is under development.

In this chapter the hardware and software elements which compose the Ictineu AUV have been described. In the short period of time since its creation, the vehicle has undergone extensive usage in many different research fields. Also, it has proved to be a very reliable platform, requiring only minor maintenance tasks. We expect this AUV to become a reference for all future prototypes developed in our laboratory.

Chapter 4

Understanding Mechanically Scanned Imaging Sonars

The purpose of this chapter is to give a brief introduction to the operational principles of MSISs by explaining the basics behind the acquisition of acoustic images as well as providing tools to understand and interpret the information they contain. Moreover, some hints about the principal issues associated to managing MSIS data are given at the end of the Chapter. Some of the examples described here were inspired by the introductory document in [Imagenex technology corp., 2002]. A deeper study on sonars and their techniques can be found in [Urick, 1983].

4.1 Principles of operation

An MSIS performs scans in a horizontal 2D plane by rotating a mechanically actuated transducer head at pre-set angular increments. For each one of the resulting angular positions, an acoustic fan shaped beam with a narrow horizontal beamwidth and a wide vertical one is produced (Figure 4.1). When this emitted acoustic signal travels through the environment and collides with any object in its path, part of the energy transmitted as a mechanical wave returns to the transducer. Measuring the time of flight of the returning wave and assuming a known value for the speed of sound in water, it is possible to determine the range at which the signal was originated. In a similar way, if the signal returning to the transducer head is analyzed for a period of time it is possible to produce not one, but a series of echo amplitude vs. range measurements. From now on, each one of these measurements will be referred to individually as a *bin*, while the set of bins obtained from a single

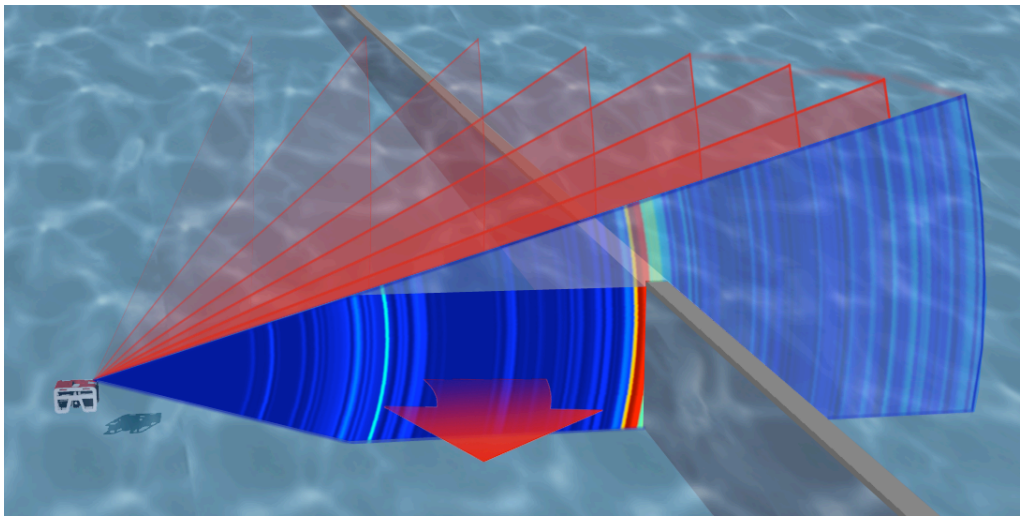


Figure 4.1: Representation of the scanning process of an MSIS.

emitted wave will be generally denominated as a *beam*. Therefore, when a transducer head oriented in a particular direction emits a pulse, a beam is produced. This beam is composed of a set of bins, each one representing the echo returning from a specific place along the transducer axis. Figure 4.2. shows an example of an image generated with real data obtained by a Tritech's Miniking imaging sonar in a shallow water trial.

4.2 Interpreting sonar images

In many cases, an acoustic image obtained in a particular scenario will closely resemble an optical image of the same place. In other cases, it may be substantially different and hence, more difficult to analyze. To interpret the information contained in an acoustic image it is necessary to understand the process behind the generation of a beam. The diagram in Figure 4.3 will serve as a guideline for the following description. The process begins with the emission of a pulse from the transducer. During the first meters, the pulse travels through the water volume without impacting with any object. Therefore, no noticeable echo is produced and only some noise is returned to the sensor head. The first significant echo return is obtained when the arc-shaped pulse reaches the bottom. Because of the large incidence angle, only a small fraction of the mechanical energy is returned and hence, the measured intensity value is small. However, as the acoustic signal advances and finds

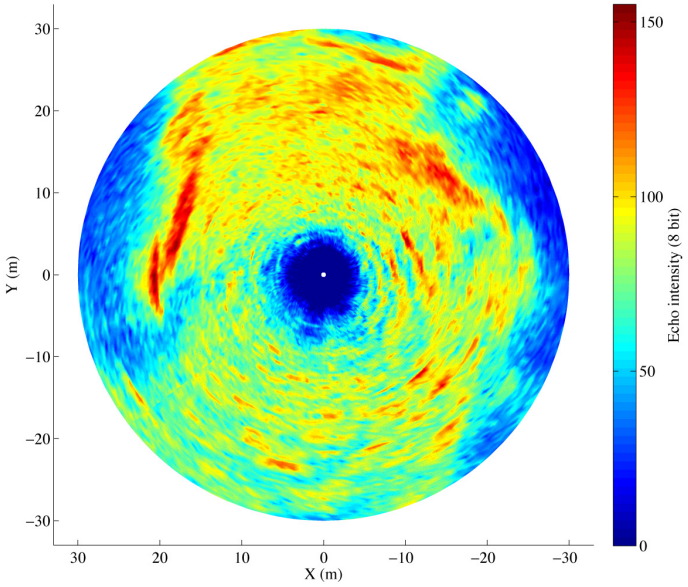


Figure 4.2: Scan obtained in a shallow water scenario.

a protruding object, an increase in the measured intensity can be observed. Notice that behind the object there is a zone where the sound can not be reflected, thus no signal is returned. This is an acoustic shadow, usually identifiable as a leak of echo intensity after an object detection. Shadows are very useful when interpreting acoustic images, since their length provides information from which the height of insonified objects can be inferred. Figure 4.4 illustrates the scanning process for an IS operating in a scenario where two objects lie on the seafbed. The image on the right represents the zones with different measured echo intensities that one should expect from an image obtained in such scenario. The largest area, in gray, corresponds to the low intensity returns from the bottom. The objects appear as high intensity zones and are represented in white. On the other hand, the absence of significant returning echoes is represented in dark color. It is worth noting how the two objects cast shadows in a similar way to which one could expect from a light source placed in the sensor head. However, the interpretation of real acoustic data is not so straightforward. First, a sonar image will always have a poor resolution due to the nature of the acoustic signals used to generate it. In addition, the materials composing the seabed will be a determining factor in obtaining information from a sonar. Generally, rough objects are better sonar targets because they return echoes in many different directions,

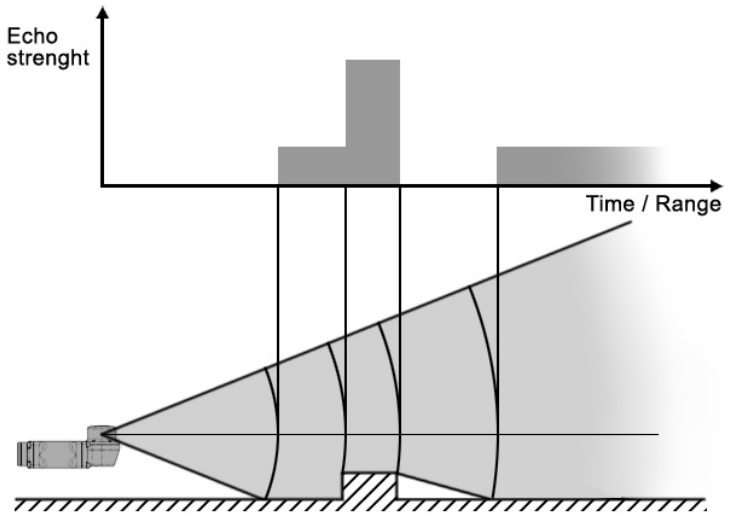


Figure 4.3: Generation of an acoustic beam.

whereas, smooth surfaces may give a very strong reflection in one particular direction, but almost none in any other. These characteristics become clear when examining the example image in Figure 4.2. Colour is assigned depending on the reception intensity level, so the zones in red represent high return areas, such as reefs or rocks; yellows and cyans represent medium/low return areas, such as the flat seabed and finally, the ones in blue represent zones from which no echo is returned. Notice that, as previously stated, shadows are found behind the high intensity zones.

It is also important to make clear that, although the sensor will reproduce in the acoustic image any tridimensional object present in the scene, it is not possible to determine its position in the vertical plane and therefore, only a 2D representation of the environment is produced. This concept is illustrated in Figure 4.5, where two objects placed at the same distance from the transducer head, but with different heights above the seabed, produce the same acoustic return. This effect is a consequence of a wide vertical beamwidth. On the one hand, it increments the capacity of the sensor to detect objects, which is convenient for some applications such as obstacle avoidance. However, it comes at a cost of introducing the indetermination in the vertical position. Moreover, errors can also affect the range measurements as a result of a wide beamwidth. As can be seen in the figure, although both objects are placed at the same radial distance from the sensor head, their linear distance along the horizontal is not the same. The resulting measured

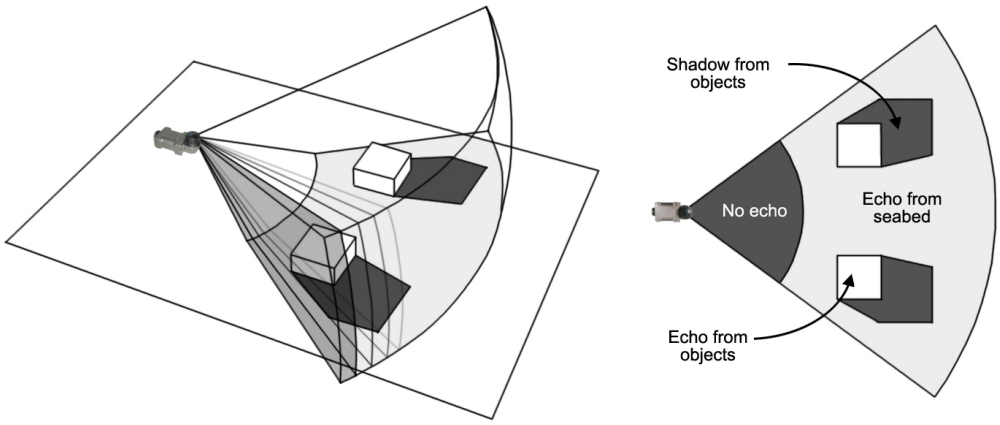


Figure 4.4: Scanning a sector to produce a sonar image.

beam however, suggests the contrary.

Another particular case is the presence of walls or other large objects in the trajectory of the emitted acoustic wave. Figure 4.6 represents this situation. Its main characteristic is that, if the obstacle is large enough, the advance of the acoustic pulse is blocked and, depending on the nature of the obstacle, part of the mechanical energy is reflected back in the opposite direction. Likewise, as the reflected pulse moves across the environment and finds other objects, part of its energy is also returned, ricochetting again with the wall and going back to the sensor head where they are interpreted as if the reflection has never taken place. In other words, the wall acts as a mirror for the acoustic pulse and, as a result, phantoms and reflections not corresponding with real objects can appear. In the example figure, the reflected pulse impacts the sensor head and produces phantom measurements in the resulting beam. This effect is usually observed when operating in confined places (for instance, a water tank); however, it is less common in larger scenarios where the reflected wave can disperse more easily. The image in Figure 4.7, which corresponds to data obtained in a small water tank, illustrates this. Reflections are easily distinguishable as vertical lines at about 12 meters from the center of the image, while the real tank boundaries are placed at about 4 meters. Note also that the small high intensity shapes placed in between actually correspond to reflections produced by the vehicle carrying the MSIS.

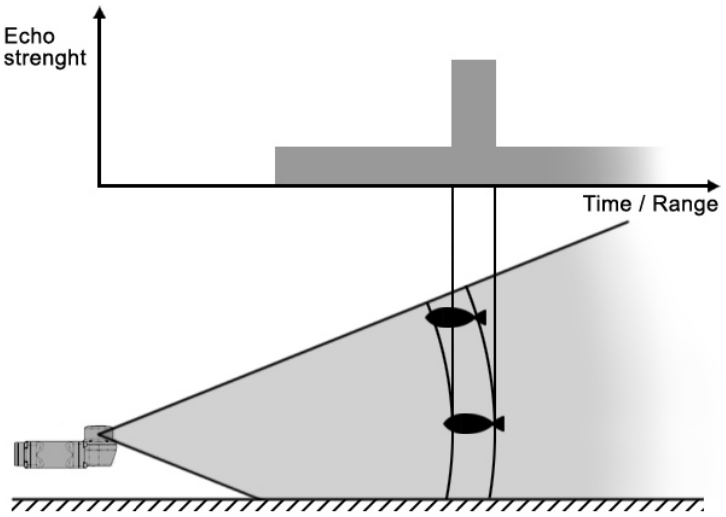


Figure 4.5: Indetermination in the vertical position of the target.

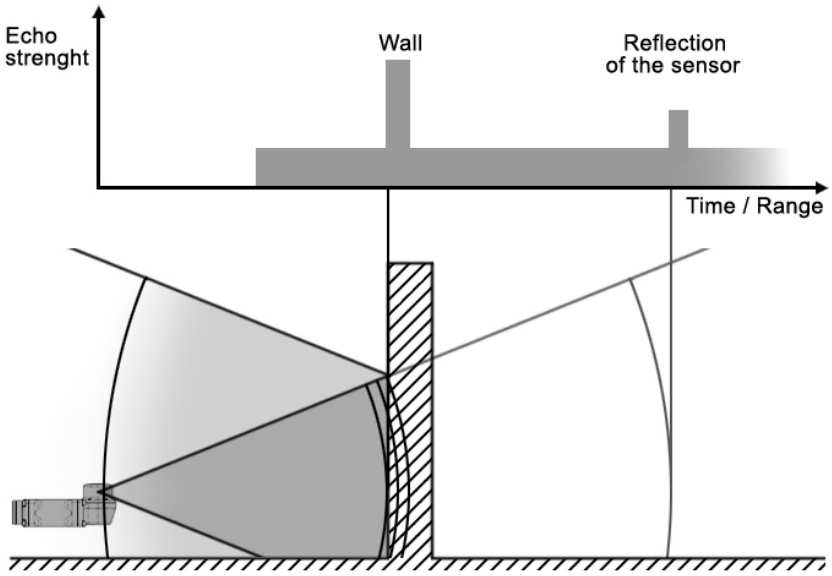


Figure 4.6: Acoustic beam reflected by a wall.

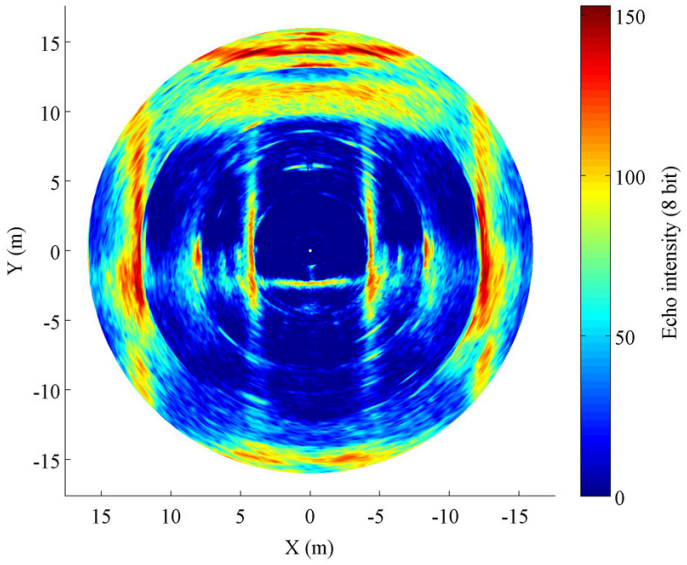


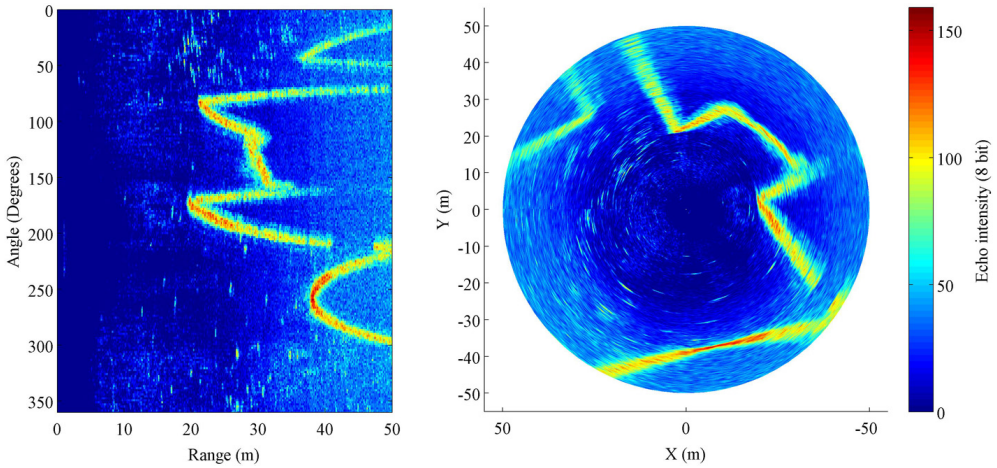
Figure 4.7: Reflections on a scan obtained in a water tank.

4.3 Peculiarities of the MSISs

The basics for acoustic image interpretation have already been introduced. There are, however, some particular characteristics of the MSISs which, although habitually overlooked, should be taken into account to obtain optimal results.

4.3.1 Polar sensor representing a Cartesian space

The acoustic data from an MSIS is usually represented as an image generated in a cartesian coordinate system because it is easier for a human observer to interpret the information it contains. However, given the nature of the measurement process, the sensor is, in fact, a polar sensor. Figure 4.8 shows the raw polar data as it is obtained from the sensor and its corresponding representation in a cartesian space. One important consequence is that, with the increment of range, a loss in the measurement resolution occurs because the bins are more dispersed as a result of the angular aperture between consecutive beams. This effect, which is inherently represented in polar, will produce gaps between the beams in the cartesian image. To avoid this, different strategies can be carried out to fill the discontinuities in the image. However, it is recommended, whenever possible, working with the raw polar measurements instead of using a cartesian image, since the change of



(a) Raw data represented in polar.

(b) The same data in cartesian.

Figure 4.8: Different representations of MSIS data obtained in a marina.

representation may alter the original data.

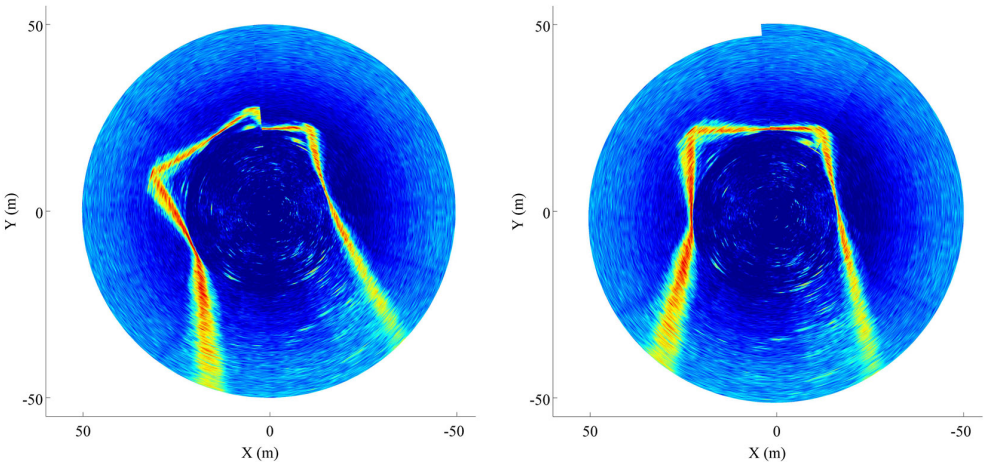
4.3.2 Continuous dataflow

The major part of the sensors typically used in localization and SLAM produce discrete amounts of information which can be treated as independent entities (e.g., images from cameras, scans from laser range finders or scans from sonar rings). This is possible because the measuring process of those sensors is very fast or even instantaneous. In the case of MSISs, the information is not gathered instantaneously but by means of a rotatory transducer head that needs a considerable amount of time to complete a turn. Since this rotation is continuous, the resulting data is not naturally split into separate subsets and hence, it is also continuous. A naive approach when working with MSIS data is to divide it into a sequence of independent 360° scan sectors. This is not an optimal procedure for many reasons. First, the division is totally arbitrary and serves no other purpose than providing a way to independently operate chunks of data. Second, the 360° scan sectors are produced at a very low rate. Finally, the first and the last beams in the scan are usually placed near each other but a considerable time lapse separates the instants in which they were obtained. As a result, when either the environment or the vehicle's position change, the generated acoustic image can eventually present a discontinuity. Better alternatives to manage continuous data include using data buffers, analyzing smaller scan sectors or even using

the data beam to beam as soon as they are measured. Some examples of operating with data from MSISs will be the subject of further discussion in Chapters 5 and 6.

4.3.3 Motion induced distortions

As stated previously, an MSIS transducer head usually needs a considerable period of time to perform a 360° rotation. In the case of the Tritech Miniking imaging sonar, the necessary minimum time to complete a scan is about 6 seconds (mechanic limit); although, depending on the settings (in particular, the range), it can increase drastically (e.g., a 50 meter range setting requires about 15 seconds to complete a scan). This is an important issue that has to be taken into account when operating with an MSIS mounted on a submersible, since the resulting acoustic images can get distorted as a consequence of the vehicle's motion. Generally, this effect can be ignored for low velocities. For higher velocities, however, it is vital to have a suitable localization method (a dynamics model, dead-reckoning sensors, SLAM, etc.) to provide the necessary position feedback to un-distort the data. It is important to note that distortions are the consequence of the combined action of both translational and rotatory movements and that their influence may vary depending on the typology of the vehicle. For instance, survey vehicles (torpedoes, flat-fish vehicles, etc.), which generally move along straight paths at considerable speeds, will be more prone to suffer translational distortions. On the other hand, hovering vehicles, like the Ictineu, generally move at lower speeds but have the capability to perform quick rotations, being more sensible to angular distortions. The image in Figure 4.9(a) shows a cartesian representation of the acoustic data obtained with the Ictineu vehicle during a test in a marina environment. Since the motion has been ignored during the generation of the image, an important distortion appears. Figure 4.9(b) presents the same dataset represented along the trajectory performed by the vehicle during the acquisition. As can be observed, when comparing it with the aerial image of the test scenario in Figure 4.9(c), the distortion in the second image is almost completely cancelled, obtaining a more accurate representation.



(a) Image generated from raw sensor data. (b) Image after undistorting the data.

(c) Zenithal view of the real scenario.

Figure 4.9: Effect of the vehicle motion on the acoustic images.

Chapter 5

Localization with an *a priori* Map

This chapter concerns the use of MSIS to solve the localization problem for an underwater vehicle navigating in a structured environment when an *a priori map* is available. The initial objective of this work was to develop a system to locate the Ictineu AUV within the squared water tank which served as the theatre of operation during the SAUC-E competition. The availability of such a localization system made possible it to pre-define a series of waypoints to be followed by the vehicle and therefore, optimize the exploration of the scenario in search of the various necessary targets to accomplish the proposed tasks. However, solving the navigation problem for an AUV moving in a water tank was not only useful for the SAUC-E competition. Further work has been undertaken to develop improved localization algorithms to work under laboratory conditions since we believe that such a system opens the door to further advanced control experiments.

Section 5.1 reviews different strategies to perform data association in localization problems, while Sections 5.2 and 5.3 present two map-based localization methods developed for the competition. The first is a simple algorithm which determines the vehicle's position by means of a voting strategy, while the second relies on an EKF to merge the information from several sensors and the tank map. A third method, which combines several aspects of the two other algorithms to improve the estimation process, is presented in Section 5.4. The chapter concludes with a summary of the advantages of the different methods and some guidelines for further work.

5.1 Data Association and Localization

A key aspect of a localization system is solving the data association, i.e. finding the correspondences between the sensor measurements and the elements contained in the map. Many authors have studied this problem with the objective of improving localization systems, but also as a necessary step to obtain a robust SLAM solution capable of relocalizing the vehicle when it gets lost or suffers large odometry errors. A particularly difficult situation is the global localization problem, also known as the "kidnapped" robot problem, where no previous estimate of the vehicle's position is available and only the information from the onboard sensors can be used to determine its position [Cox, 1991]. The techniques to perform this data association can be divided between those that analyze the pose space and those that analyze the correspondences between the measurements and the map.

In the first approach, a set of candidate vehicle positions is considered and rated accordingly to the evidence presented by the sensor measurements. The hypothesis presenting a better consistency between the measurements and the map is the one that corresponds most to the vehicle's real position. Different methods are used to represent the space of possible vehicle positions. One example is the Monte Carlo localization [Fox et al., 1999; Thrun et al., 2004], which addresses the problem by sampling a set of random vehicle locations covering the entire area and computing the likelihood of each position. An alternative is using grid sampling, such as Markov localization [Fox et al., 1998; Burgard et al., 1996], where the map is represented as an occupancy grid in which each cell represents a particular vehicle position. The cost of this kind of algorithm is proportional to the size of the map (number of particles, or cells in the grid).

In the second approach to the data association problem, the process consists of defining a set of hypotheses for the pairings between the sensor measurements and the features in the map. The hypothesis with the largest number of consistent matches should define the correct vehicle position. The cost of this method depends on the size of the correspondence tree. To limit the complexity of the search, different approaches have been developed such as the use of simple geometric constraints [Grimson, 1990], the hipothesize and test technique [Lim and Leonard, 2000], branch and bound algorithms [Castellanos and Tardós, 1999], graph theory [Bailey et al., 2000], random sampling [Neira et al., 2003] and voting [Paz et al., 2005].

In the context of the present work, different examples can be found for the localization of underwater vehicles operating in structured environments. In the Autonomous System Laboratory of the University of Hawaii, range measurements gathered with a set of fixed-bearing sonar beams were used to

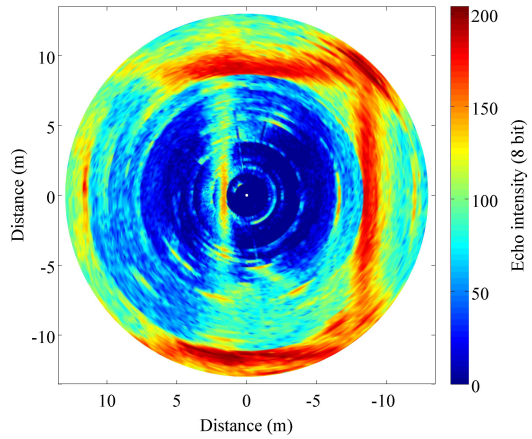
update a Kalman filter and estimate the position of the vehicle in a water tank [Nie et al., 1998]. Using this system, and thanks to its omni-directionality, Odin AUV can navigate keeping its relative orientation with respect to the walls of the water tank. A more elaborate system is described in [Caccia et al., 2001] where a profiling sonar is used to track the walls of the water tank and hence the robot is allowed to change its heading freely. In previous work, our team solved this problem using a coded pattern, lying on the bottom of a water tank together with a real-time vision system able to provide accurate absolute position estimates at 12 Hz [Carreras et al., 2003]. An example of a real application can be found in [Kondo et al., 2006], where a localization system makes the autonomous inspection of a breakwater possible.

5.2 Voting-Based Localization Method

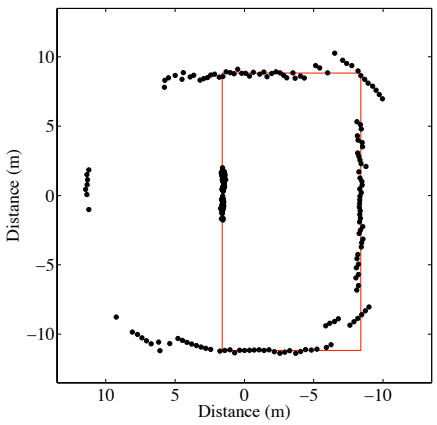
This localization system described in the following, was initially conceived as a method to determine the position of the Ictineu AUV within the SAUC-E water tank during the initialization phase of the Kalman filter-based localization algorithm that will be presented in Section 5.3. However, preliminary tests showed its potential and we soon realized that it could become a localization system on its own. Moreover, the fact that this method was already required as part of the alternative algorithm together with the strict time constraints set by the competition, prompted the team to finally implement it into the vehicle's software architecture while making the necessary minor changes to convert the method into a full localization algorithm. Due to its simplicity of operation and proved reliability, it became our final choice for positioning the vehicle during the competition.

5.2.1 Voting Algorithm

The method presented here determines the vehicle's position by exploring the correspondence between the measurements and the elements of the scenario using a voting-based strategy. This algorithm only requires an *a priori* map and the measurements from an MSIS, a compass and a pressure sensor to locate the vehicle within a particular environment. The map, the MSIS and the compass are used to determine the vehicle's heading and position in the horizontal plane while the pressure sensor is sufficient to estimate its vertical position as the measurements provided by the sensor are directly related with the depth at which the vehicle is operating. For the purpose of this work, only square-shaped scenarios were taken into account to test the algorithm. However, there are no particular restrictions limiting the use of this algorithm



(a) Acoustic image obtained in the SAUC-E water tank during the competition.



(b) Resulting set of bins after selecting the ones with the highest intensity (tank walls in red).

Figure 5.1: Selection of the most representative bins.

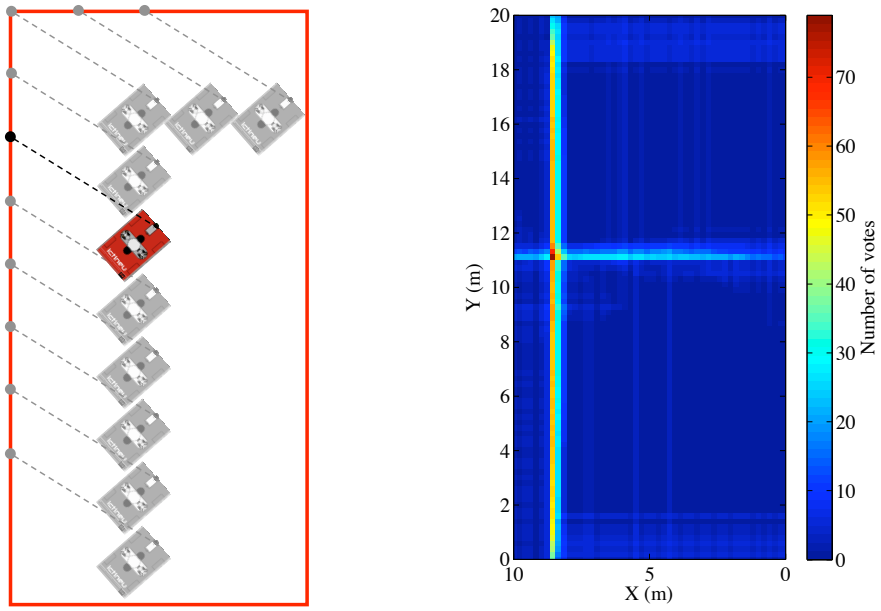
to such scenarios and it is assumed that the algorithm should be able to work in more complex environments.

The acoustic image in Figure 5.1(a) illustrates the type of data we can expect when working in a water tank. Note that the range of the sensor has been set to approximately half of the longest tank dimension, which makes it possible to observe a great part of the tank from most of the vehicle's positions while avoiding the appearance of reflections in the image. The objects (walls) present in the vicinity of the sensor appear in the image as elongated zones populated with high intensity echo returns (shapes in red). In order to reduce the number of bins involved in the process and as a consequence, to improve the overall computational efficiency, only those bins which are the global maximum of each beam are selected. Moreover, a threshold is applied to select the ones with an intensity value high enough to correspond with the detection of a real object, discarding the less significant ones. In Figure 5.1(b) the selected bins appear as small black dots. For reference, the real water tank boundaries are represented by a rectangle in red. The majority of the selected bins are expected to match the real position of the water tank walls. Therefore, it is possible for a set of bins, regardless of ghosts and reflections in the acoustic data, to reach a consensus and determine the true position of the tank limits and reciprocally, localize the vehicle. In the present algorithm, this consensus is determined by means of an adapted version of the Hough transform [Duda and Hart, 1972; Ballard, 1987]. The

classical implementation of the Hough transform is a voting algorithm for line feature extraction used in digital image processing. However, for this particular application, it has been modified to identify the position of the vehicle inside the water tank area. The first step of the procedure is using the information from the *a priori* map to define a discretized space representing the environment (tessellation of the water tank area). The goal of this gridded model of the water tank is to accumulate evidence regarding the actual vehicle position in the form of votes. At the end of the voting process, the cell with the highest number of votes will be chosen as the most likely to correspond with the current vehicle position. The set of high intensity bins previously selected are responsible for determining where the votes should be assigned in the voting space. Assuming that the orientation of the water tank is known and that the vehicle is equipped with a compass, it is possible to obtain the angle between the vehicle and the tank. Furthermore, the position of any bin with respect to the vehicle's frame is totally determined by the range and bearing measurements provided by the sensor. Therefore, for each single bin and given all this information, a search for compatible vehicle locations is carried out. Every candidate position should accomplish two conditions. First, the bin associated with the position must overlap the boundaries of the tank at some point. Second, the vehicle must be placed within the limits of the tank. The compatible vehicle positions that meet the two conditions describe a particular L-shaped zone which can be easily determined. Figure 5.2(a) depicts a schematic representation of this process. In the voting space, each one of the cells corresponding to the described locus will receive one vote. If this is repeated for all the selected high intensity bins from a complete scan, the accumulation of votes will result in a voting space such as the one represented in Figure 5.2(b). In the example, the cell with the highest number of votes appears in dark red and matches the real vehicle position, as can be observed by comparing the result with the central point on the original scan in Figure 5.1(a).

5.2.2 Dealing with Continuous Acoustic Images

In order to determine the position of the vehicle, this algorithm needs to have a sufficient amount of information available. It is not unusual to find situations in which the vehicle's vantage point causes limited observation of the scenario. For this reason, extracting high intensity bins from a 360° scan sector is usually a good approach. Another important aspect is the frequency at which those scan sectors are obtained. As stated in Chapter 4, the range setting of the sensor is one of the factors directly related with the



(a) Locus of all the possible vehicle positions assuming that the measured bin corresponds with a tank wall.

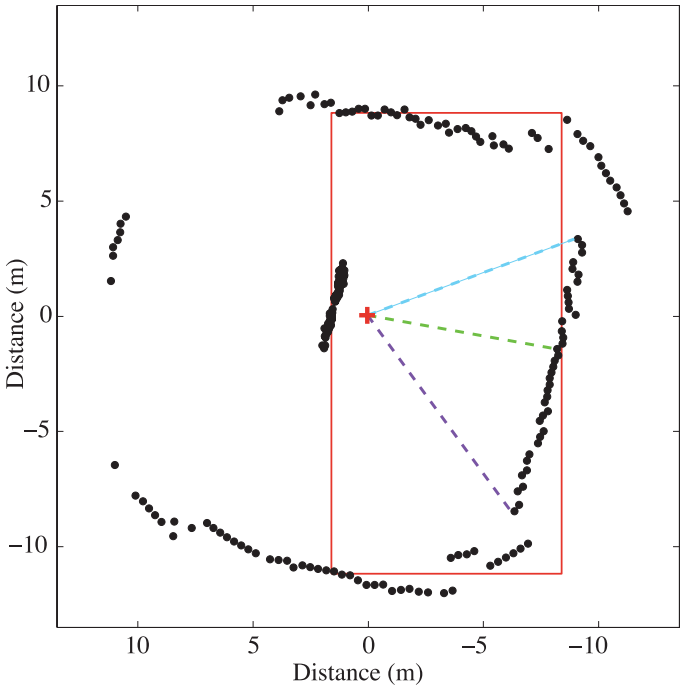
(b) Resulting voting space after assigning all the votes produced by a set of bins from a complete 360° scan.

Figure 5.2: Voting process for vehicle localization.

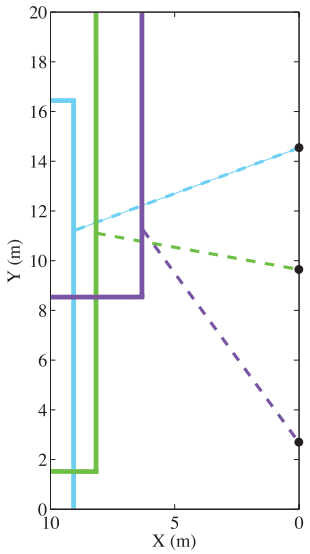
amount of time necessary to produce a complete scan. A good option is to choose the shortest range possible (about half of the longest dimension of the tank). However, even when working at its shortest range configuration, MSISs usually need a considerable period of time to obtain a scan (e.g. the Miniking has a lower time limit of about 6 seconds). To summarize, using 360° scan sectors for the voting process is desirable but usually obtaining such scans takes a lot of time and, as a result, the vehicle position estimates can only be produced at a low frequency. In order to overcome this disadvantage, a simple modification has been implemented in the algorithm: the new beams produced by the sensor are accumulated by means of a data buffer so, at any moment, the information from the most recent 360° scan can be recalled to produce a new votation and determine the actual position of the vehicle. As new areas are explored with each beam arrival, new bins are added to the data buffer. Simultaneously, older bins falling outside the considered scan sector are removed. In other words, instead of interpreting the sonar measurements as a discrete sequence of consecutive"acoustic images", they are treated as a continuous dataflow from which is possible to instantaneously, and at any desired frequency, recover the most recent "snapshot" of the environment.

5.2.3 Managing Compass Errors

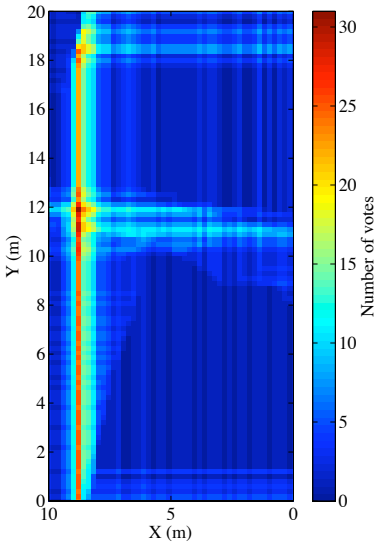
Another important modification of the present algorithm is a method to adapt the voting process to possible angular errors. As mentioned before, a compass is used to determine the current vehicle orientation with respect to the water tank. However, one should expect magnetic disturbances to affect the sensor when the vehicle navigates through scenarios where the presence of ferro-magnetic elements is not uncommon (e.g. wire meshes inside the concrete walls of a water tank). An error in the estimation of this angle will be transmitted to the localization process as an incorrect allocation of the votes in the voting space. The example in Figure 5.3(a) represents a set of bins affected by this error (black dots), that are misaligned with respect to the real water tank position (red rectangle). The relative positions of three particular bins and the vehicle have been represented with dashed lines of different colors. In the scheme on Figure 5.3(b), the places that will receive the votes for each one of those three bins have been represented with solid lines. Note that since all three bins correspond to the same wall on the right, the three vertical sections should overlap. However, instead of overlapping, the votes are spread over three parallel vertical zones as a result of the misalignment caused by the angular error. Figure 5.3(c) shows the resulting voting space after using all the selected bins, with the the major part



(a) The angular error causes a misalignment between the measured bins and the expected position of the tank.



(b) The effect of voting with misaligned bins.



(c) Resulting voting space after voting with misaligned data.

Figure 5.3: Voting with an angular error.

of the votes spread over a wide area. Of course, such a space cannot produce a reliable position estimate. The strategy to address this issue is simple and effective. First, it is assumed that some angular error is affecting the entire set of voting bins. The exact value of this error is unknown but is assumed to be bounded. Then, instead of performing the votation on a single space, several spaces are taken into account. Those spaces are identical, except for the fact that different error values are used to correct the relative vehicle-map angular measurements employed during the voting in each space. As a result, a set of voting spaces with different error assumptions are obtained (those in Figures 5.2(b) and 5.3(c) can be taken as examples of the kind of spaces that are obtained). Next, a search for the vehicle position estimate is carried out, first by selecting the candidate with the maximum number of votes from each one of the different spaces and then choosing the overall most voted candidate (see Figure 5.4). The reason behind this process is simple. A great accumulation of votes implies that the bins are perfectly aligned with the boundaries of the water tank. Therefore, the voting space that contains the candidate position with the maximum number of votes has to be the one with the angular error assumption that best matches the discrepancy in the real system. For instance, the winning candidate position in Figure 5.3(c) has only 31 votes as a result of a bad angular error assumption that is not able to correct the misalignment in the voting bins. This contrasts with the space obtained in Figure 5.2(b), where a correct assumption in the error is able to produce a winner with 79 votes. It is worth noting that working with multiple voting spaces is computationally less efficient than performing the voting for all the error hypotheses in a single space. However, using a single space would offer a more dispersed solution as a consequence of placing votes with wrong hypotheses. Moreover, without the possibility of discriminating the effect of combined false assumptions from the accumulation of votes due to the correct hypothesis, this method would present a considerable risk of producing false position estimates.

5.2.4 Discretization of the Voting Space

One of the key issues while implementing the voting algorithm is to select a correct grid resolution when defining the discretized voting space. Choosing a large cell size will reduce the resolution of the space and hence, the precision of the measured positions. On the other hand, choosing a smaller cell size increases the computational cost of the algorithm, especially when dealing with angular errors, as several voting spaces are used simultaneously. A good initial assumption is choosing a grid resolution comparable to the

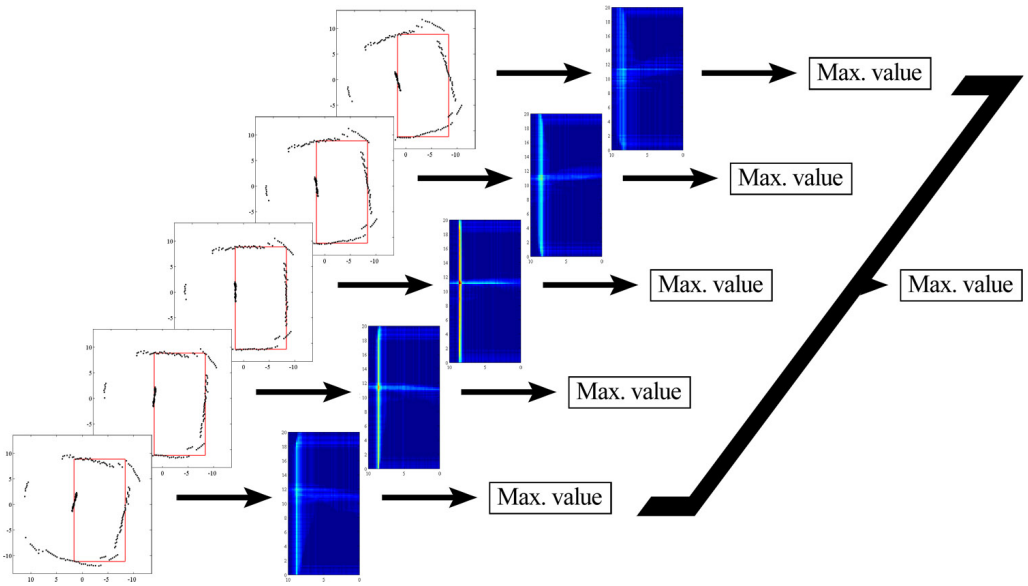


Figure 5.4: Voting in multiple spaces with different angular error hypotheses.

actual resolution of the acoustic images generated by the MSIS (distance between consecutive bins). However, there are situations were taking a smaller resolution will have some benefits. One of these situations is when the vehicle is moving at a high speeds and the bins gathered during a 360° scan are obtained from significantly different positions. Consequently, when a voting is performed with those bins, their votes are placed accordingly with the position of the vehicle at the moment when the measurement took place. This is reflected in the voting space as a dispersion of the votes along the vehicle's path and produces an unpredictable outcome in the final position estimate. It is worth noting that the present algorithm cannot estimate the vehicle's motion during the acquisition of the bins and thus, it is not possible to correct this issue in the voting process. However, if the resolution of the voting space is reduced, the uncertainty produced by this effect can be enclosed in a bigger cell size. Of course, position measurements will also have lower resolution but will benefit from a steadier behavior.

5.2.5 Experimental Results

In this section, two examples of the algorithm running in real environments are presented. The first corresponds to an experiment executed under laboratory conditions that will be used throughout this chapter as a benchmark to compare the different localization methods. The second corresponds to

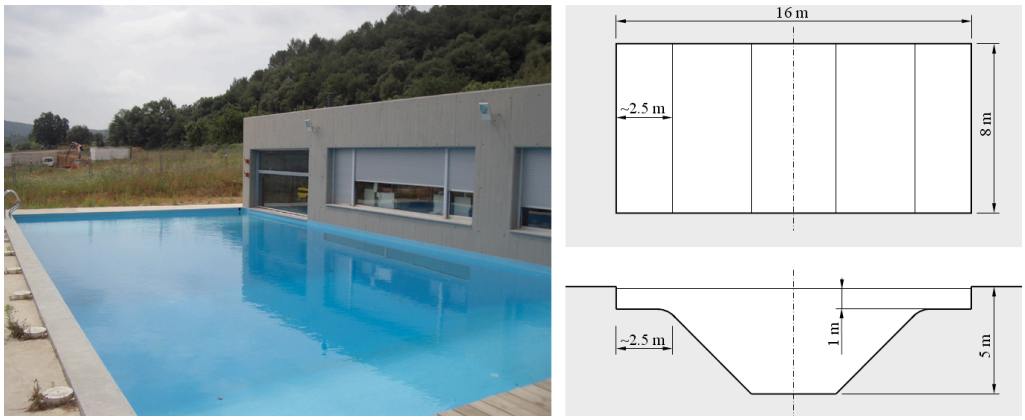


Figure 5.5: CIRS water tank at the Unversity of Girona.

real data obtained during the final run of the SAUC-E competition.

CIRS Water Tank Test

This experiment took place in the water tank at the *Centre d'Investigació en Robòtica Submarina* (CIRS) at the University of Girona (Figure 5.5). Although the Garbi vehicle was used to obtain the dataset, it can be considered as equivalent to one obtained with the Ictineu vehicle since both share compatible sensor suites. The dataset is formed by measurements from the MSIS and the DVL (which includes a compass and a pressure sensor, as explained in Section 3.6.2). The MSIS was set to a range of 10 m and a resolution of 0.1 m (100 bins per beam) and was able to produce a complete 360° scan sector in about 6.6 s (0.15 Hz). During the experiment, the vehicle was operated to perform a trajectory in the deepest part of the water tank. Starting from the center, it went to a side, near to a water tank wall and then described a roughly square-shaped loop. The algorithm was set to perform votations with 360° scan sectors at a frequency of about 0.3 Hz (i.e. each time a new 180° scan sector was obtained). The chosen resolution of the discretized space was 0.1 m. Figure 5.6 shows the measured positions (red dots) and the resulting estimated trajectory (line in black). No ground truth was available to validate the results; however, they seem consistent with the actual path followed by the vehicle. Although the algorithm exhibits a great level of reliability, a single erroneous position measurement was obtained during the execution of the algorithm. This was mainly because of the sonar readings produced by the slanted walls limiting the deepest zone of the water tank.

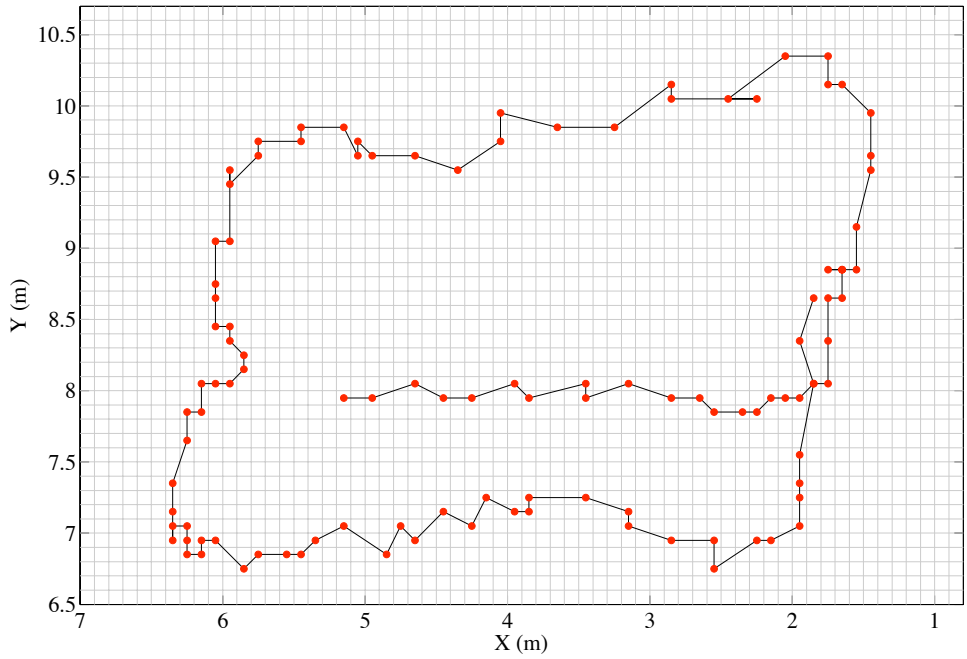


Figure 5.6: Trajectory obtained with the votation-based localization algorithm for the experiment performed in the CIRS water tank.

During the voting, the algorithm expects the selected bins to correspond with the outer limits of the water tank and, as no alternative strategy was defined for such measurements, they are erroneously assigned producing a false position estimate. Tipically, the bins corresponding with the slanted walls represent a small portion of the total, so the correctly placed votes prevail and generally, a correct position is produced. Nevertheless, erroneous position measurements are not an issue as they are infrequent and are easily detected as outliers when comparing their distance to previous position estimates.

SAUC-E Final Run

The results presented in this section were produced with sensor data gathered by the Ictineu AUV during its performance in the SAUC-E 2006 competition final. The run took place in the water tank of the Underwater Stage at the Pinewood Studios in Buckinghamshire (United Kingdom) and consisted of a sequence of tasks to be accomplished autonomously by the vehicle

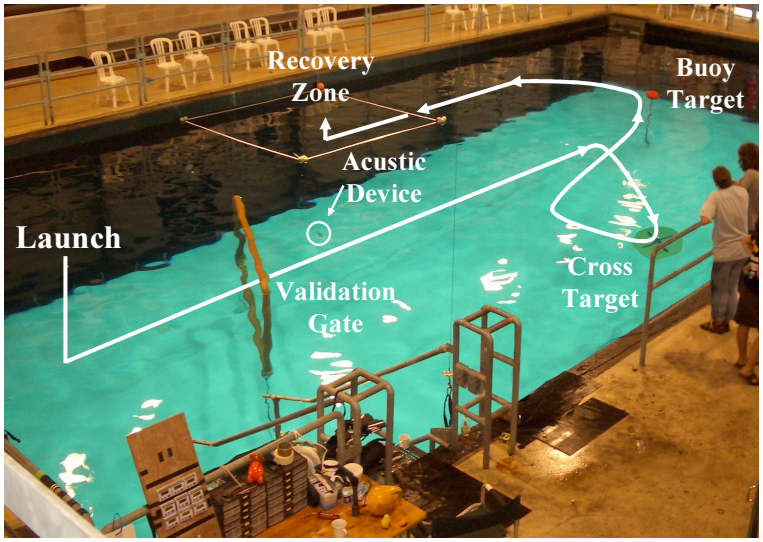


Figure 5.7: Water tank at the Underwater Stage of the Pinewood Studios with all the elements conforming the setup for the SAUCE competition.

(Figure 5.7). First, the vehicle started from the release point and had to submerge and pass through a validation gate. Then, a cross target lying on the bottom of the tank had to be found and a marker dropped over it. A second target, an orange buoy, had to be located and impacted with the frame of the vehicle. Finally, the vehicle had to end the mission by surfacing at a designated recovery zone marked by an acoustic device. During the final run, the Ictineu AUV attempted all four tasks. With two of them partially achieved and two more successfully completed, our team got the final victory.

A version of the votation-based localization algorithm was implemented in the Ictineu AUV in order to provide position feedback to the software architecture during the competition. To reduce the cost of the algorithm and free resources for other tasks that were running simultaneously, the voting space grid resolution was set to 0.5 m. This decrease in the precision did not represented any problem, as the purpose of the localization system was to drive the vehicle to particular zones of the water tank rather than positioning it in a precise spot. Moreover, the reduction in cost made it possible to increase the system's output rate to a frequency of 2 Hz. Unfortunately, it was not possible to record the trajectory estimation obtained by the algorithm during the execution of the mission. In spite of this, the data loggers were able to record the measurements from all the sensors and thus, the estimated path could be post-processed. Figure 5.8 represents the resulting

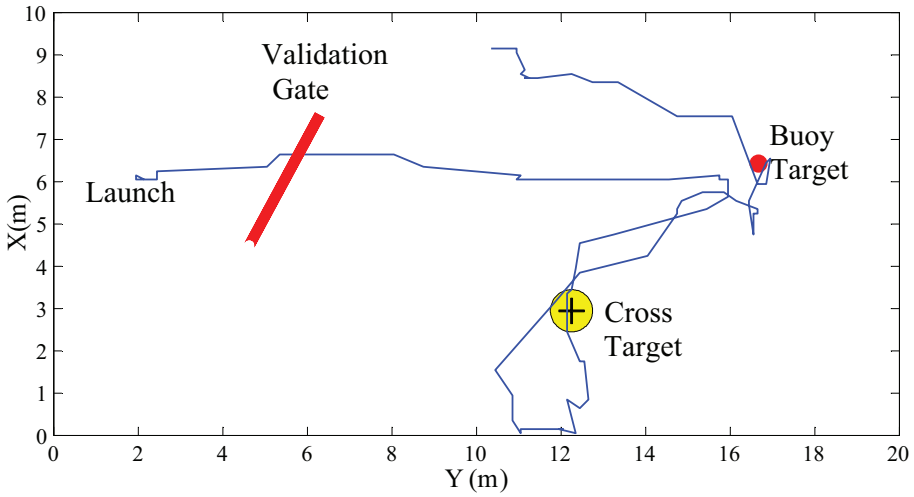


Figure 5.8: Trajectory performed by the Ictineu AUV during the final run of the SAUC-E 2006.

trajectory. It is worth noting that a voting space resolution of 20 cm had been used during the generation of this trajectory in order to represent more accurately the path actually followed by the vehicle during the run. In fact, it is accurate enough that it is possible to observe how the vehicle moves through the validation gate, hovers over the cross target to release a marker, attempts impacting the buoy and finally moves towards the recovery zone. For reference, the estimated trajectory can be compared with the one appearing in the video in [Ictineu, 2007], which corresponds with the actual run performed by the Ictineu during the final.

5.3 EKF-Based Localization Method

This second proposal relies on an EKF to estimate a state vector containing the position and velocity of the vehicle. A simple kinematic model allows predicting the state at any moment while the information from different sensors is used to update it. In addition to the compass and the pressure sensor, a DVL sensor is incorporated to provide direct measurements of the vehicle's velocities. The MSIS together with an *a priori* map produce information regarding the absolute position of the vehicle within the environment. This approach defines the *a priori* map as a set of line features representing the planar objects present in the scenario (in the application at hand, the four

vertical walls limiting a water tank). Then, the readings from the MSIS are analyzed beam to beam in order to determine their bin with the highest echo-intensity return. Again, the assumption that this bin should correspond to a feature in the map is taken. Therefore, after determining the bin-line feature correspondence, the discrepancy between their positions in the space is used to update the current vehicle position estimate.

5.3.1 Defining the Map

The *a priori* map of the environment $\mathcal{M}$ is defined as a set of n line features:

$$\mathcal{M} = \{\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_n\}.$$

Each line describe the horizontal cross section of a vertical planar structure (e.g. walls of the water tank as seen from a zenithal point of view) and are represented by their *rho-theta* pair as:

$$\mathbf{l}_n = [\rho_n \, \theta_n]^T$$

which represents the perpendicular distance from a line feature to a common base reference B. The position of this base reference is arbitrary. However, placing it in one of the corners of the water tank – aligned with the direction of the walls– is usually is a good choice, as the definition of line parameters would be straightforward. The origin is placed at the water level and the Z axis is looking down. In other words, anything placed below the water surface receives a positive value on its Z coordinate. It is worth noting that the β angle representing the orientation of the base reference B (i.e. orientation of the water tank) with respect to an earth fixed reference frame E must be known when working with a compass in order to integrate their measurements. This issue will be discussed in Sections 5.3.3 and 5.3.5. A representation of the map and the different reference frames can be seen in Figure 5.9

5.3.2 State Vector

The state vector contains information regarding the position and velocity of the vehicle at time k:

$$\mathbf{x}(k) = [x \; y \; z \; \psi \; u \; v \; w \; r]^T$$

where, following the nomenclature proposed in [Fossen, 1994], the vector $[x \; y \; z \; \psi]$ represents the position and heading of the vehicle in the local base

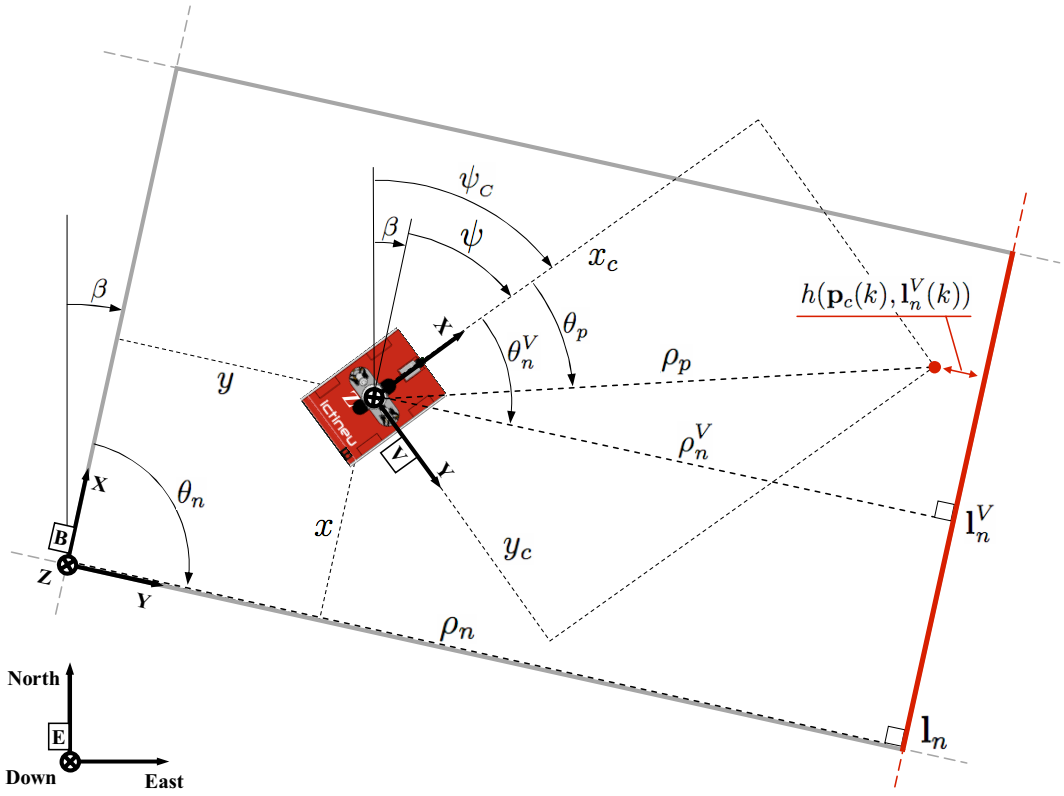


Figure 5.9: Representation of the different elements involved in the localization method.

reference B, while $[u\ v\ w\ r]$ are their corresponding linear and angular velocities which are represented on the vehicle's coordinate frame V. The angles and velocities for the *roll* and *pitch* DOFs are not included in the state vector as all our vehicles are passively stable in those DOFs.

5.3.3 Initializing the Filter

The initial value of the state vector $\mathbf{x}(0)$ should be estimated before starting the Kalman filter. As mentioned previously, the voting algorithm from Section 5.2 was originally designed for this task. The process should begin with the obtention of a complete 360° scan with the MSIS and the corresponding angular measurements from the compass. After defining a voting space accordingly to the *a priori* information of the scenario, this set of measurements is used to perform a voting. The winning candidate will determine the situa-

tion of the vehicle within the tank. Therefore, it can be used to initialize the estimated XY vehicle position in the state vector. As this value represents a measurement relative to a fixed base reference, some level of uncertainty has to be considered. One alternative to assign this initial value is using a filter to estimate the position of the vehicle with the measurements that have been previously associated in the voting. However, the results obtained with this method tend to be optimistic. Here, a simpler alternative has been chosen. The uncertainty is set accordingly to the precision of the voting process by taking into account the grid resolution used during the estimation of the initial position. Although taking this resolution to set the standard deviation of the position is a good approach, adopting a slightly larger value is usually recommended in order to cope with unexpected errors. Obtaining the initial values for the depth and the heading of the vehicle is much easier as they can be directly measured with the outputs from the pressure sensor and compass. This time, the variance of this initial estimate will be determined by the sensor's accuracy. It should be noted that the vehicle heading ψ is referenced to the base reference B, while the angle measured from the compass is obtained with respect to magnetic north. Thus, the angle between the earth-fixed reference E and the base reference B must be taken into account when using the compass measurement to initialize the vehicle's heading. During the initialization phase, the vehicle is presumed to be static, as it would make the voting process to determine the vehicle position more reliable. Therefore, the velocities can be set to zero, along with their uncertainty, as a perfect knowledge of the variables can be assumed. The resulting estimate for the state vector at time 0 is:

$$\hat{\mathbf{x}}(0) = \begin{bmatrix} x_{_V} \\ y_{_V} \\ z_{_P} \\ \psi_{_C} - \beta \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{P}(0) = \begin{bmatrix} \sigma^2_{_V} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{_V} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{_P} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{_C} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where the subindex V stands for the voting algorithm, P for pressure sensor, C for compass and β corresponds to the angle between the north and the base reference B as represented in Figure 5.9.

5.3.4 System Model

A simple 4 DOF constant velocity kinematics model is used to predict how the state will evolve from time $k-1$ to time k:

$$\begin{aligned}
\mathbf{x}(k) &= f(\mathbf{x}(k-1), \mathbf{n}(k-1)), \\
\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_{(k)} &= \begin{bmatrix} x+(uT+n_u\frac{T^2}{2})cos(\psi)-(vT+n_v\frac{T^2}{2})sin(\psi) \\ y+(uT+n_u\frac{T^2}{2})sin(\psi)+(vT+n_v\frac{T^2}{2})cos(\psi) \\ z+wT+n_w\frac{T^2}{2} \\ \psi+rT+n_r\frac{T^2}{2} \\ u+n_uT \\ v+n_vT \\ w+n_wT \\ r+n_rT \end{bmatrix}_{(k-1)}
\end{aligned} \tag{5.1}$$

where $\mathbf{n} = [n_u \; n_v \; n_w \; n_r]^T$ represents a vector of white Gaussian acceleration noises with zero mean. They are additive in the velocity terms and propagate through integration to the position. The covariance of the $\mathbf{n}$ vector is represented by the system noise matrix $\mathbf{Q}$:

$$\begin{aligned}
E\left[\mathbf{n}(k)\right] &= \mathbf{0}, \qquad E\left[\mathbf{n}(k)\mathbf{n}(j)^T\right] = \delta_{kj}\mathbf{Q}(k), \\
\mathbf{Q} &= \begin{bmatrix} \sigma_{n_v}^2 & 0 & 0 & 0 \\ 0 & \sigma_{n_u}^2 & 0 & 0 \\ 0 & 0 & \sigma_{n_w}^2 & 0 \\ 0 & 0 & 0 & \sigma_{n_r}^2 \end{bmatrix}.
\end{aligned}$$

The model described in (5.1) is non-linear and therefore, the prediction should be performed with the EKF equations (see AppendixA). This version of the filter linearizes the system model around the current estimate with the Jacobian matrices $\mathbf{F}$ and $\mathbf{W}$:

$$\mathbf{F}(k) = \frac{\partial f}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k), \mathbf{0}) = \begin{bmatrix} 1 & 0 & 0 & -\hat{u}T\sin\hat{\psi}-\hat{v}T\cos\hat{\psi} & T\cos\hat{\psi} & -T\sin\hat{\psi} & 0 & 0 \\ 0 & 1 & 0 & \hat{u}T\cos\hat{\psi}-\hat{v}T\sin\hat{\psi} & T\sin\hat{\psi} & T\cos\hat{\psi} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{W}(k) = \frac{\partial f}{\partial \mathbf{n}}(\hat{\mathbf{x}}(k), \mathbf{0}) = \begin{bmatrix} \frac{T^2}{2}\cos\hat{\psi} & \frac{-T^2}{2}\sin\hat{\psi} & 0 & 0 \\ \frac{T^2}{2}\sin\hat{\psi} & \frac{T^2}{2}\cos\hat{\psi} & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & 0 & \frac{T^2}{2} \\ T & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \end{bmatrix}.$$

5.3.5 Measurement Model

The vehicle is equipped with a number of sensors providing direct observations of particular elements of the state vector and hence, a linear observation model can be used (see Appendix A). The general model for such measurements is written in the form:

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k|k-1) + \mathbf{m}(k)$$

where $\mathbf{z}$ is the measurement vector and $\mathbf{m}$ represents a vector of white Gaussian noises with zero mean affecting the observation process. The covariance of the $\mathbf{m}$ vector is represented by the measurement noise matrix $\mathbf{R}$:

$$E\left[\mathbf{m}(k)\right] = \mathbf{0}, \qquad E\left[\mathbf{m}(k)\mathbf{m}(j)^T\right] = \delta_{kj}\mathbf{R}(k).$$

The form of the observation matrix $\mathbf{H}$ changes according to the measurements obtained from the sensors. Hereafter, different forms of the H matrix are presented for each of the cases that can take place in our particular system:

Velocity: A DVL sensor produces velocity measurements in the 3DOF. Assuming a sensor coordinate frame coincident with the vehicle reference V, or at least, a known transformation relating them, the velocity measurements can be taken as direct observations of the vehicle's velocities. Thus, the observation matrix $\mathbf{H}$ can be written as:

$$\mathbf{H}_D = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \tag{5.2}$$

and its measurement noise as:

$$\mathbf{R}_D = \begin{bmatrix} \sigma_{Du}^2 & 0 & 0 \\ 0 & \sigma_{Dv}^2 & 0 \\ 0 & 0 & \sigma_{Dw}^2 \end{bmatrix}.$$

Depth: The measurements from a calibrated pressure sensor can be easily operated to obtain an estimation of the vehicle's depth (position in the Z axis). The resulting **H** matrix is:

$$\mathbf{H}_P = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{5.3}$$

The variance of the depth measurement will be represented by:

$$\mathbf{R}_P = \sigma_P^2.$$

Heading: The compass measures the angle of the vehicle with respect to magnetic north. As the vehicle heading ψ in the state vector is referenced to the base frame B, the angle β (angle of the frame with respect to the north) has to be subtracted from the compass reading to produce the measurement $\mathbf{z}_C$. The resulting angle can be used to update the state vector with the following **H** matrix:

$$\mathbf{H}_C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{5.4}$$

As the β angle is perfectly known as part of the *a priori* map, subtracting it from the compass measurement has no effect on the uncertainty. Therefore, the measurement noise $\mathbf{R}_C$ is represented by the variance of the compass:

$$\mathbf{R}_C = \sigma_C^2.$$

It is worth noting that, depending on the configuration of the system, different readings could happen simultaneously. For instance, the DVL sensor in the Ictineu vehicle (see description in Section 3.6.2) is also equipped with a compass and a pressure sensor. Therefore, each time a new output is produced by the device, not only the velocities should be updated but also the depth and heading estimates in the state vector. In order to deal with multiple measurements simultaneously, a composed form of the **H** matrix can be obtained by adding different rows from (5.2), (5.3) and (5.4):

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_D \\ \mathbf{H}_P \\ \mathbf{H}_C \end{bmatrix}, \qquad \mathbf{R} = \begin{bmatrix} \mathbf{R}_D & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_C \end{bmatrix}.$$

Another particularity of the system is that the DVL sensor is able to measure the velocity with respect to the ground as well as the relative velocity between the sensor and the water volume below. The first set of measurements are worthwhile as they represent a direct estimation of the movement

of the vehicle with respect to a fixed reference. Unfortunately, some scenarios are particularly difficult (those in shallow water or where reverberations take place) and such velocity measurements are incorrect. On the other hand, water volume velocities do not seem to be so dependent on the environment as the ground ones. For this reason, although they are less accurate, they can be used to estimate the vehicle's motion. However, this can only be done in those situations where the water volume is static, or in other words, when no water currents are present. The Argonaut DVL has the capacity to determine the quality of the received signals automatically and provides a status value for the velocities. Accordingly, the measurements with a bad status are discarded before the update by removing the corresponding rows in (5.2).

5.3.6 Updating the Position Estimate

The process of estimating the vehicle's position from the integration of velocity measurements suffers from an inherent drift. To deal with this, the MSIS is used together with the *a priori* map $\mathcal{M}$ to correct the absolute vehicle position in the state estimate. This process is carried out each time a new single beam is obtained from the imaging sonar and begins with the selection of the bin with the maximum intensity value. This measurement represents a point in the space which is the most likely to evidence the presence of an object in the scene and, as a consequence, to correspond with a line feature in the *a priori* map. The information regarding the point-line pairing will be used to perform an update in the state estimate from the filter (see Figure 5.9).

The selected high intensity bin from the MSIS is produced in polar coordinates:

$$\mathbf{p}_p(k) = [\rho_p(k), \theta_p(k)]^T, \quad \mathbf{p}_p(k) = \hat{\mathbf{p}}_p(k) + \mathbf{u}(k)$$

where $\mathbf{u}$ is a zero mean white Gaussian noise affecting the sensor during measurement:

$$E[\mathbf{u}(k)] = \mathbf{0}, \qquad E\left[\mathbf{u}(k)\mathbf{u}(j)^T\right] = \delta_{kj}\mathbf{P}_p(k), \qquad \mathbf{P}_p(k) = \begin{bmatrix} \sigma^2_{\rho_p} & 0 \\ 0 & \sigma^2_{\theta_p} \end{bmatrix}.$$

To determine its correspondence with the line features composing the map the first step is transforming the bin parametrization from polar to cartesian coordinates.

$$\begin{aligned} \hat{\mathbf{p}}_c(k) &= q(\hat{\mathbf{p}}_p(k)) \\ \begin{bmatrix} \hat{x}_c(k) \\ \hat{y}_c(k) \end{bmatrix} &= \begin{bmatrix} \hat{\rho}_p(k)\cos\hat{\theta}_p(k) \\ \hat{\rho}_p(k)\sin\hat{\theta}_p(k) \end{bmatrix}. \end{aligned}$$

The Jacobian of the non-linear q function is also obtained for further calculations concerning the measurement uncertainty.

$$\mathbf{J}_q = \frac{\partial q}{\partial \mathbf{p_p}}(\hat{\mathbf{p}}_p(k)) = \begin{bmatrix} \cos\hat{\theta}_p(k) & -\hat{\rho}_p(k)\sin\hat{\theta}_p(k) \\ \sin\hat{\theta}_p(k) & \hat{\rho}_p(k)\cos\hat{\theta}_p(k) \end{bmatrix}.$$

For the sake of simplicity, the sensor reference frame has been taken as coincident with the vehicle's reference frame V. Otherwise, the appropriate transformations (see Appendix B) should be carried out to represent the bin in cartesian coordinates in the V reference frame. The next step is to represent each one of the line features $\{\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_n\}$ stored in the $\mathcal{M}$ map in the same V frame so they can be compared with the selected bin. The g function obtains the parameters for a line feature $\mathbf{l}_n$, originally defined in B, with respect to the position of the vehicle's frame stored in the current state estimate $\hat{\mathbf{x}}(k|k-1)$:

$$\begin{aligned} \hat{\mathbf{l}}_n^V(k) &= g(\mathbf{l}_n, \hat{\mathbf{x}}(k|k-1)), \\ \begin{bmatrix} \hat{\rho}_n^V \\ \hat{\theta}_n^V \end{bmatrix} &= \begin{bmatrix} \rho_n - \hat{x}\cos\theta_n - \hat{y}\sin\theta_n \\ \theta_n - \hat{\psi} \end{bmatrix}. \end{aligned}$$

As the map is assumed to be perfectly known, the uncertainty depends only on the vehicle state estimate $\hat{\mathbf{x}}$. Therefore, only the Jacobian of the g function with respect of the state is necessary:

$$\mathbf{J}_g = \frac{\partial g}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k|k-1)) = \begin{bmatrix} -\cos\theta_n & -\sin\theta_n & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

With both the sonar return and the map in the same reference frame, an implicit non-linear measurement function h stating that the distance among them is zero and thus, that the point belongs to the line, can be defined (see Figure 5.9) [Castellanos and Tardós, 1999]:

$$\begin{aligned} h(\mathbf{p}_c(k), \mathbf{l}_n^V(k)) &= \rho_n^V - x_c\cos\theta_n^V - y_c\sin\theta_n^V, \\ &= 0, \end{aligned}$$

with:

$$\begin{aligned} \mathbf{H}_1 &= \frac{\partial h}{\partial \mathbf{l}_n^V}(\hat{\mathbf{p}}_c(k), \hat{\mathbf{l}}_n^V(k)) = [1 \quad \hat{x}_c\sin\hat{\theta}_n^V - \hat{y}_c\cos\hat{\theta}_n^V], \\ \mathbf{H}_2 &= \frac{\partial h}{\partial \mathbf{p}_c}(\hat{\mathbf{p}}_c(k), \hat{\mathbf{l}}_n^V(k)) = [-\cos\hat{\theta}_n^V \quad \sin\hat{\theta}_n^V], \end{aligned}$$

where $\mathbf{H}_1$ and $\mathbf{H}_2$ are the Jacobians of the implicit measurement function h with respect to the selected bin and a particular line of the map.

Multiple hypotheses can be made relating the bin with the different n line features. To produce the update, one of them has to be chosen as valid. For this purpose, an Individual Compatibility (IC) test is performed for each hypothesis by means of the presented measurement equation as:

$$\begin{aligned} \mathbf{S} &= \mathbf{H}_1\mathbf{J}_g\mathbf{P}(k|k-1)\mathbf{J}_g^T\mathbf{H}_1^T + \mathbf{H}_2\mathbf{J}_q\mathbf{P}_p(k)\mathbf{J}_q^T\mathbf{H}_2^T, \\ D^2 &= h(\hat{\mathbf{p}}_c(k), \hat{\mathbf{l}}_n^V(k))^T\mathbf{S}^{-1}h(\hat{\mathbf{p}}_c(k), \hat{\mathbf{l}}_n^V(k)) < \chi^2_{d,\alpha}. \end{aligned} \tag{5.5}$$

Distance D^2 is the Mahalanobis distance [Mahalanobis, 1936]. The correspondence is accepted if the distance is less than $\chi^2_{d,\alpha}$, with α defined as the confidence level and $d = \dim(h)$. The Nearest Neighbor (NN) selection criterion determines that, among the pairings that satisfy (5.5), the one with the smallest Mahalanobis distance is chosen and the association hypothesis is accepted. If none of the pairings pass the test, the bin is considered as spurious and rejected. This association process ensures not only the correct association of the measurement but also allows rejecting spurious data from the sonar image, such as those produced by multipath or by the presence of other objects not represented in the map.

Having the data association solved, an update of the vehicle state estimate can be performed using the EKF equations for an implicit measurement function [Castellanos and Tardós, 1999]:

$$\begin{aligned} \mathbf{K} &= \mathbf{P}(k|k-1)\mathbf{J}_g^T\mathbf{H}_1^T\mathbf{S}^{-1} \\ \hat{\mathbf{x}}(k) &= \hat{\mathbf{x}}(k|k-1) - \mathbf{K}\, h(\hat{\mathbf{p}}_c(k), \hat{\mathbf{l}}_n^V(k)) \\ \mathbf{P}(k) &= (\mathbf{I} - \mathbf{K}\,\mathbf{H}_1\mathbf{J}_g)\mathbf{P}(k|k-1) \end{aligned}$$

5.3.7 Experimental Results

This section presents results obtained with the EKF-based localization method for the CIRS water tank test previously presented in Section 5.2.5. In addition to the MSIS, compass and pressure sensor measurements already used in the previous method, the velocity measurements from the DVL are also employed. As mentioned in Section 3.6.2, the DVL can produce velocity measurements with respect to the ground and to the water. The reduced dimensions and the reflectivity of the concrete walls in the CIRS water tank make it a complex scenario in which to operate the DVL sensor. In consequence, a considerable number of measurements receive a bad quality status and are rejected. Although bottom tracking velocities are generally more precise, water velocities are more reliable and the number of rejected measurements is lower. During this test, both velocity measurements were used

to update the filter in order to obtain a better estimate. It is worth noting that using the water velocity measurements to estimate the vehicle movement is only possible when operating in scenarios were no currents exist and the water volume can be assumed as static. On the contrary, the system model should be adapted to account for the effect of water currents.

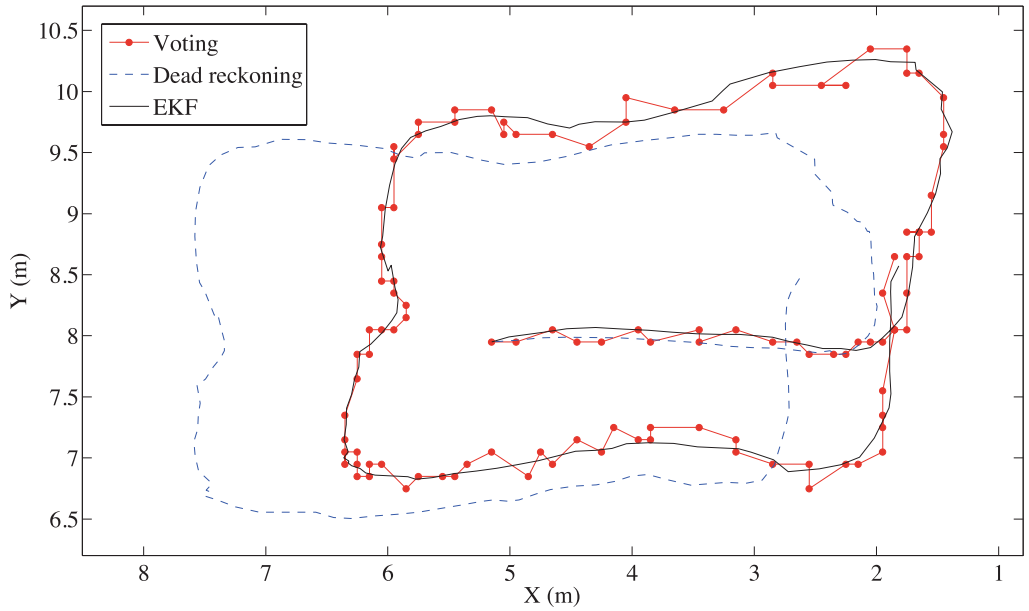


Figure 5.10: Trajectory obtained with the EKF-based localization algorithm compared with trajectories obtained with other methods.

Figure 5.10 represents the trajectory obtained with the present EKF-based localization method (line in black). It can be observed that this trajectory is consistent with the one obtained with the previous method (line in red). For comparison purposes, the dead reckoning trajectory obtained by running the filter without performing the position updates with the MSIS measurements is also represented (dashed line in blue). As the position in the dead reckoning estimate is obtained only by integrating the velocity measurements, the process is inherently affected by drift. This effect is even more noticeable as a result of the important perturbations affecting the velocity measurements when operating in such an adverse scenario. On the other hand, when the MSIS measurements are contrasted with the *a priori* map and used to update the estimate, information regarding the absolute posi-

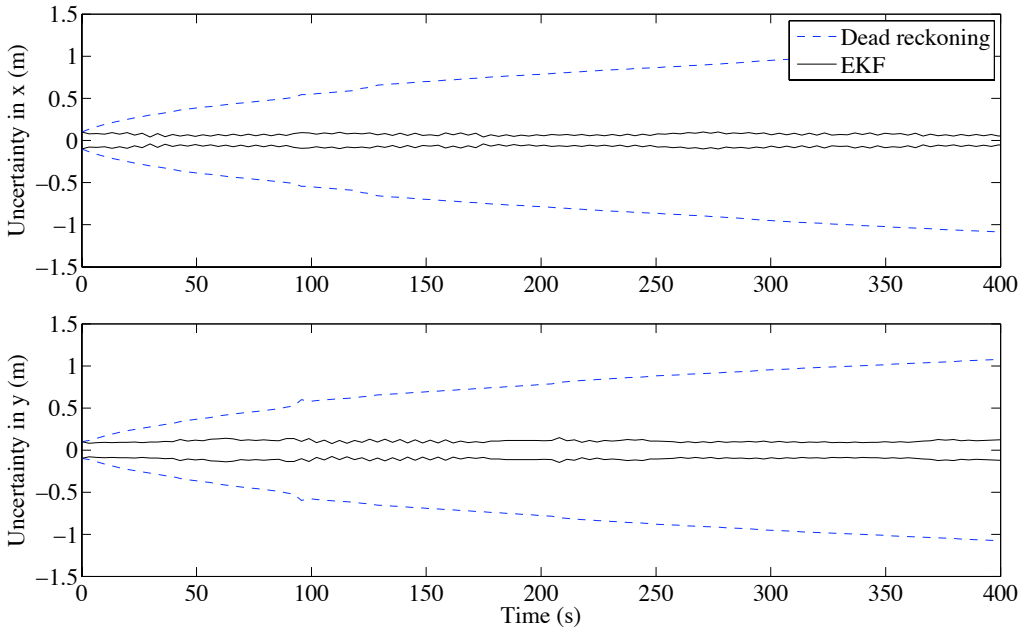


Figure 5.11: Uncertainty of the estimated position in the B reference frame represented by its 2σ bounds.

tion within the tank is incorporated and, as a result, the drift disappears. As shown in Figure 5.11, the effects of these updates are also reflected in the error plots of the position estimate. The represented 2σ bounds grow without limit in the dead reckoning estimate (dashed line in blue) as a result of the velocity errors integration, while they remain constant when updated with the map (black line).

5.4 Hybrid approach

This third approach to the localization problem is an attempt to merge the best from the two methods already presented. The main advantage of the votation-based localization is the capability of producing independent absolute measurements. This means that even when a position estimation fails, posterior votations can relocalize the vehicle since this is a global localization method that does not rely on position tracking. On the contrary, the above mentioned EKF-based method corrects the position estimate in the filter by means of relative measurements between the vehicle and a wall from the *a priori* map and since this correction relies on the current position estimate,

errors can eventually cause the vehicle to get lost and force the system to re-initialize. This second method however, has its strongest point in the use of an EKF which allows improving the position estimate by integrating information from additional sensors and, at the same time, makes it possible to obtain position estimates at a higher rate with a fraction of the computational cost required for the voting algorithm.

The proposed hybrid method includes an EKF update with the measurements from a DVL, a compass and an adapted version of the presented voting algorithm. The benefits are twofold. First, the position estimate will now be corrected with absolute measurements, making it more reliable and avoiding eventual track loses. Second, the position estimates from the filter can be included in the voting process to avoid the dispersion of votes along the vehicle's trajectory while moving.

5.4.1 The Filter

The EKF used in this hybrid approach is equivalent to the one presented in Sections 5.3.2 to 5.3.5 and only differs in the process to update the position estimate. Instead of using the procedure described in Section 5.3.6, this method uses the output from the adapted voting algorithm that will be presented in Section 5.4.2. The measurement $\mathbf{z}_V$ provided by this voting algorithm corresponds to the coordinates of the vehicle's position within the 2D map of the water tank. Since the measurements are obtained in the B reference frame, they can be directly integrated by means of the following linear measurement equation:

$$\begin{aligned} \mathbf{z}_V(k) &= \mathbf{H}_V \mathbf{x}(k|k-1) + \mathbf{m}(k), \\ \mathbf{H}_V &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \end{aligned}$$

where $\mathbf{m}$ represents a vector of white Gaussian noises with zero mean affecting the measurement process. The covariance of the measurement is represented by $\mathbf{R}_V$:

$$E\left[\mathbf{m}(k)\right] = \mathbf{0}, \qquad E\left[\mathbf{m}(k)\mathbf{m}(j)^T\right] = \delta_{kj}\mathbf{R}_V(k).$$

Although the voting algorithm has proved to be quite reliable, there still are situations where reduced visibility, phantoms or reflections in the acoustic data can affect the process and cause errors in the resulting measurement. Since these values are direct observations of components in the state vector, integrating these erroneous measurements strongly affects the correctness of the estimate. To avoid this, we can determine if a given position measurement

from the voting algorithm is consistent with the current predicted vehicle position estimate by means of the innovation term ν_V, which represents the discrepancy between them. Its value and covariance are:

$$\begin{aligned} \nu_V &= \mathbf{z}_V - \mathbf{H}_V \hat{\mathbf{x}}(k|k-1), \\ \mathbf{S}_V &= \mathbf{H}_V P(k|k-1)\mathbf{H}_V^T + \mathbf{R}_V. \end{aligned}$$

To determine the compatibility of the measurement, a IC test is performed. The measurement can be considered as corresponding to the current position estimate if the Mahalanobis distance D^2 satisfies:

$$D^2 = \nu_V^T \mathbf{S}_V^{-1} \nu_V < \chi^2_{d,\alpha},$$

where $d = \dim(\mathbf{H}_V)$ and α is the desired confidence level. All the measurements passing the IC test are then used to update the filter. Since the measurement model equation is linear, this can be done with the classic KF update equations that can be found in Appendix A.

5.4.2 Adapted Voting Algorithm

As introduced in Section 4.3.3, distortions affecting the acoustic data could be differentiated between those resulting from rotations and displacements. In the method presented in Section 5.2, using the information from the compass to determine the relative orientation between the vehicle and the scenario implicitly takes into account rotations during the votation and removes the effect of its distortion. On the other hand, distortions due to the displacement of the vehicle are ignored since it is assumed that the vehicle is stationary or moving very slowly. In the present method, such an assumption can be removed by integrating the position estimate from the EKF into the voting process. The process begins by tagging each new beam arriving from the MSIS with the current vehicle position in the state vector and storing them in the data buffer. Before producing a votation, a set of n selected high echo intensity bins from a 360° scan sector is obtained together with its corresponding set of vehicle positions $\{[x_1, y_1], \ldots [x_n, y_n]\}$. This concept is shown graphically in Figure 5.12, where the position of the vehicle at the beginning of the scan (acquisition of the n bin) is described by $[x_n, y_n]$ and the current vehicle position (bin number 1, the most recent one) is represented by $[x_1, y_1]$. The voting method described in Section 5.2 places the votes in accordance with the position of the vehicle at the moment when the measurement of each particular bin took place. The vehicle's movement results in a dispersion of the votes along the performed trajectory which can

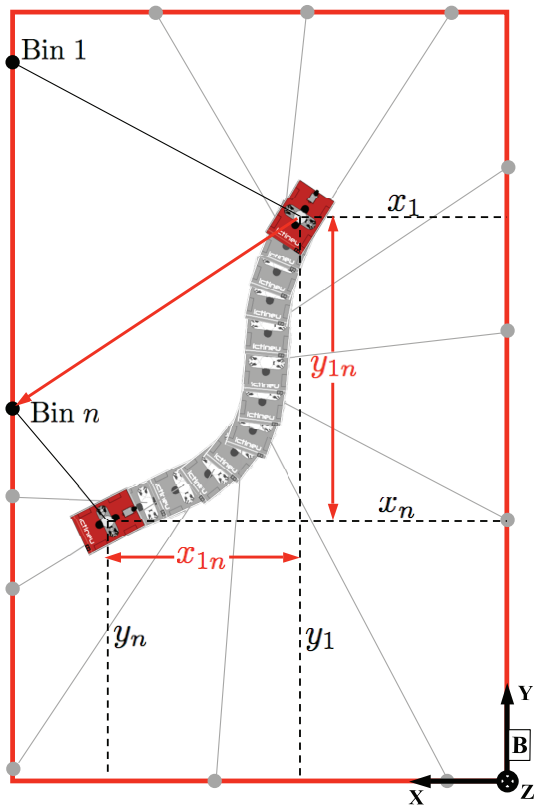
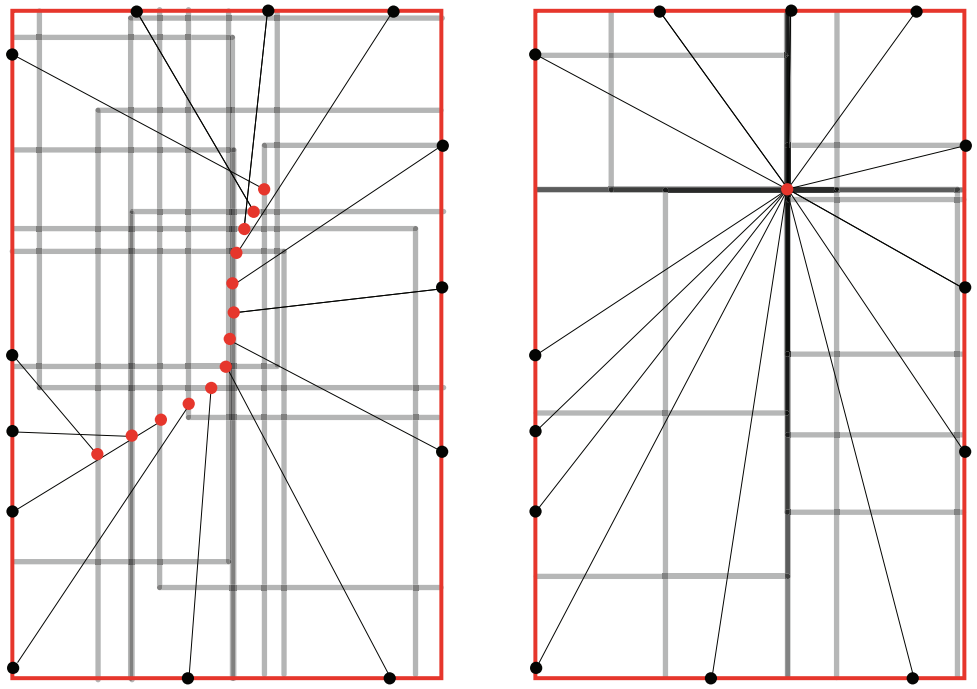


Figure 5.12: Compensating the effect of motion during the voting process

produce a loose estimation of the real vehicle position (see Figure 5.13(a)). In the present method, this effect is compensated for by using the stored vehicle positions to determine the relative positions of the different bins and making them vote for the current location of the vehicle instead of voting for all the different positions along the trajectory (see Figure 5.13(b)). In the example in Figure 5.12, the difference between the current position and the position at which the n bin was obtained is represented by $[x_{1n}, y_{1n}]$. Composing the vehicle's referenced location of the n bin with this relative measurement makes determining the position of that same bin with respect to the current vehicle location (represented by a red arrow) possible and therefore, assigns the votes accordingly. Transforming all the votes from the whole set of n bins so they can vote for a common vehicle position removes the effect of motion-induced distortions. Moreover, the resulting voting space presents a better description of the real position because of a more focused distribution of votes. Figure 5.14 shows two voting spaces generated with the voting algorithm from Section 5.2 (on the left) and the present one (on



(a) Each bin places the votes according to its corresponding vehicle position.

(b) All the bins vote for the current vehicle position.

Figure 5.13: Compensating the motion-induced dispersion of votes.

the right). Although both of them were generated from the same set of sonar measurements, it can be seen how the first one shows a sparse distribution of votes, making it difficult to discern the winning position, while the second one has a small cluster with a high concentration of votes which is easy to identify as the winner.

5.4.3 Experimental Results

Again, the CIRS water tank experiment is used to test this new proposal. Figure 5.15 represents the trajectory obtained with the current approach (black line), which strongly resembles the ones obtained with the previous methods: votation-based (line with dots in red) and EKF-based (green line). For comparison purposes the dead-reckoning trajectory (dashed line in blue) has been also included. The algorithm has been set to perform updates with the information from the adapted voting algorithm, which produces measurements at a frequency of about 0.3 Hz. These measurements are also represented in

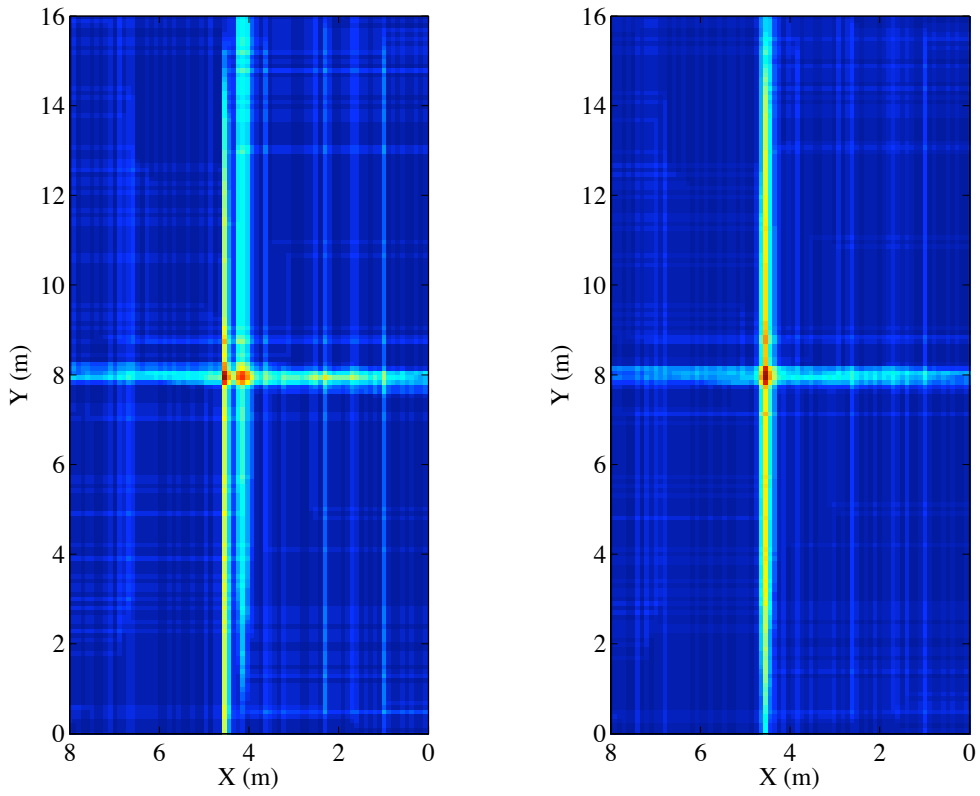


Figure 5.14: Comparison between the original voting algorithm (left) and the adapted version (right).

the figure (black crosses) so they can be compared with those obtained with the original voting algorithm (red dots). As can be seen, the measurements are quite similar, which is reasonable since the experiment was performed at a low velocity and the induced distortions were not important. In some parts of the trajectory, the improvement in the voting spaces obtained with the new method is considerable (see the example shown in Figure 5.14). However, in other zones, where perturbances affect the measurements from the DVL, the improvement is hardly appreciable. This is because the effect of the corrections performed during the voting process is dependent on the quality of the estimated trajectory. The side effects of performing absolute position updates are the elimination of the drift affecting the estimate done with the dead-reckoning sensors and the bounding of the vehicle position errors. Figure 5.16 represents the uncertainty of the estimated position error for the current method (in black) represented together with the uncertainty

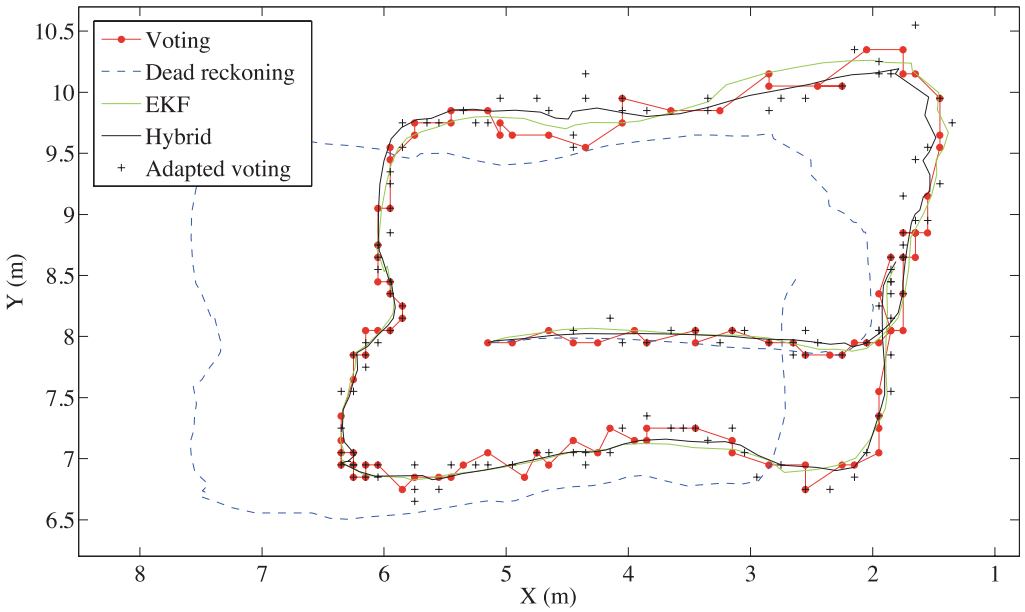


Figure 5.15: Trajectory obtained with the hybrid localization algorithm compared with trajectories obtained with the other methods.

of the dead-reckoning error obtained by running the filter without the absolute position updates (blue dashed line). As expected, the first one remains constant (except for small peaks produced by the absence of reliable DVL measurements) while the second shows the typical unbounded error growth.

5.5 Summary and Further Work

In this chapter, three different localization methods have been presented. The first, the voting-based algorithm, provides absolute position measurements and combines simplicity and reliability. Angular distortions affecting the acoustic data are reduced by integrating compass measurements. Distortions caused by displacements are ignored under the hypothesis that the vehicle is static or moving very slowly. Of course, the reliability of the method is affected when this hypothesis does not hold. One weak point of the method is that the computational cost is directly related with the resolution and frequency at which the measurements are obtained. During the SAUC-E

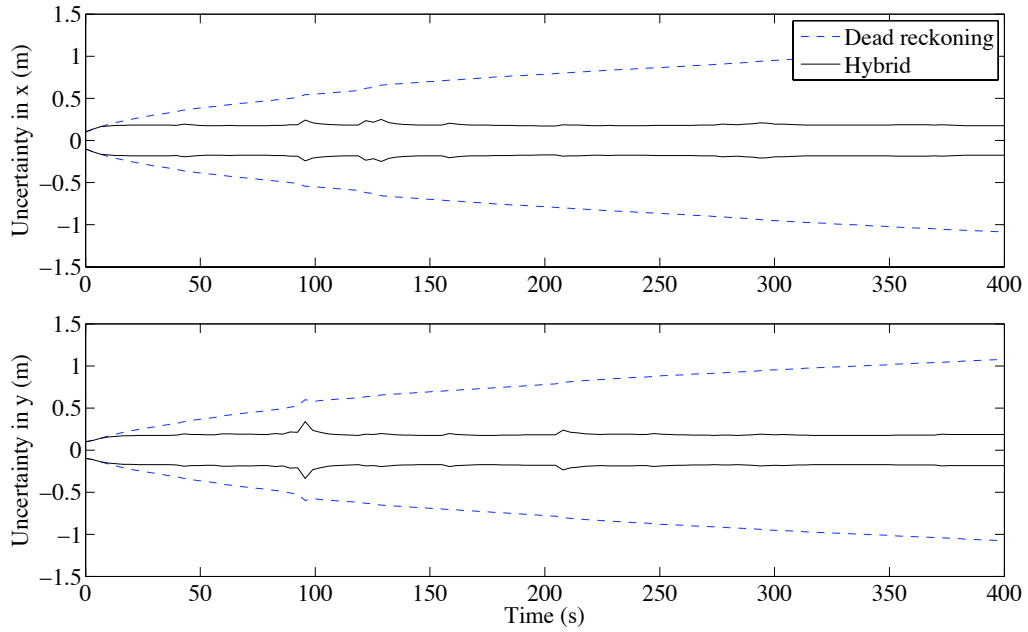


Figure 5.16: Uncertainty of the estimated position in the B reference frame represented by its 2σ bounds.

competition, the version of the algorithm running in the vehicle was set to work at a low resolution in order to increase the measurement rate and fulfill the real time requirements. This is the only one of the three presented methods implemented in real time in the Ictineu software architecture and tested under working conditions. The second method relies on an EKF for position estimation. Its strong point is the use of the filter itself, which makes it possible to constantly estimate the vehicle's state and merge information from different sensors. The MSIS data is treated beam to beam to produce corrections with the *a priori* map. As the vehicle's position is taken into account for each individual beam, the effect of distortions is implicitly corrected. The main disadvantage of this method is that using a single bin to perform the updates does not provide enough information to determine the absolute vehicle position and, therefore, if something fails (incorrect association, absence of measurements, etc), the vehicle can eventually get lost. In fact, this disadvantage is corrected in the third presented method. Again, an EKF is used but, this time, the algorithm relies on absolute position measurements obtained from a modified voting algorithm to perform the updates. The modified voting procedure uses the estimated position to remove the effect of motion in the acoustic data and thus, to improve the reliability

of the whole algorithm even when the vehicle is moving fast. Further work includes implementing this third method in the Ictineu software architecture so it can be tested under real working conditions in the CIRS water tank.

One limitation common to all the algorithms is that, although the vehicle can move in 3D, only 2D maps are considered. This is not an issue if the elements composing the map correspond to vertical planes in the real scenario. On the contrary, the map will not be valid for a vehicle navigating at different depths since, in this situation, the position of the sensed non-vertical objects could change considerably with respect to their original description in the map. An illustrative example can be found in the CIRS water tank (see Figure 5.5). The different methods take into account only the outermost limits of the tank, composed of vertical walls, but ignore the presence of the two slanted walls placed near the center. This is sufficient to localize the vehicle because, as a consequence of the large vertical beamwidth of the MSIS, the limits are visible even when the vehicle is navigating at a few meters under the water. Moreover, the slanted walls appear in the acoustic data as zones with lower intensity values than those from the outer boundaries and therefore are generally discarded in the high intensity bin selection step. Removing this limitation in the description of the scenario, either by integrating 3D maps or by using a set of 2D maps defined for different working depths, would make it possible to use non-vertical structures to add more information to the system thereby improving the quality of the measured position. In the CIRS water tank, for instance, it is not possible for the MSIS to detect the boundaries completely when the vehicle is navigating close to the bottom in the deepest zone of the tank. The slanted walls, however, appear in the acoustic data and, if a compatible map is defined, can be used to determine the vehicle's position.

Chapter 6

Simultaneous Localization and Mapping

In this chapter a SLAM framework for AUVs equipped with an MSIS operating in manmade structured environments is proposed. In the previous chapter, the use of techniques such as the Hough transform and the Kalman filter have been studied in the context of a localization problem. Here, these techniques are further explored for their application in SLAM. The proposed approach is composed of two parts running simultaneously. The first one is a line feature extraction algorithm which is in charge of managing both the measurements arriving from the MSIS and the vehicle position estimates from the SLAM system to look continuously for new features by means of a voting scheme. Eventually, when a new feature is detected, the algorithm also estimates its uncertainty parameters through the analysis of the imprint left in the acoustic images. The second part is a Kalman filter implementation which is the core of the proposed SLAM system. This filter merges the information from various sensors (DVL, compass and pressure sensor) and the observations from the feature extraction algorithm in order to estimate the vehicle's motion and to build and maintain a feature based map (see Figure 6.1 for a diagram of the complete system). In addition, the problems associated with large scenarios have also been addressed through the implementation of a local map building procedure. At the end of the chapter, two tests performed with real sensor data endorse the proposed SLAM approach. The first one employs the dataset corresponding to the previously presented CIRS water tank test, while the second undertakes a more realistic application scenario with a dataset obtained in an abandoned marina.

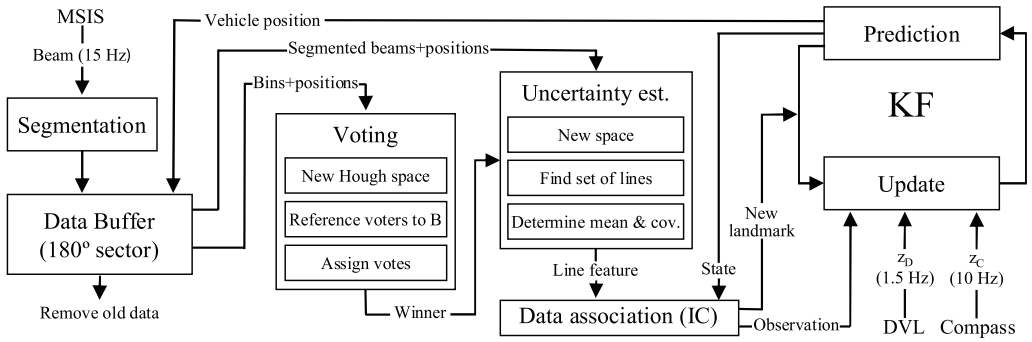


Figure 6.1: Diagram of the proposed SLAM approach.

6.1 Line Feature extraction

There is a significant published bibliography on segmentation, classification, registration and feature extraction from acoustic images [Daniel et al., 1998; Tena et al., 1999; Reed et al., 2001]. Generally, those works are focused on dealing with natural environments from which features corresponding to compact regions with high intensity backscatter were extracted. In most of the underwater SLAM approaches reviewed in Chapter 2, these regions are usually modeled as point landmarks. The process often requires dealing with significant background noise and extracting supplementary characteristics regarding their size and shape [Tena, 2001] or even their associated shadows [Reed et al., 2001] to improve the discriminability. Nevertheless, working with landmarks extracted from acoustic data is not an easy task because their appearance may change substantially when observed from different vantage points. The problem for structured scenarios like the ones handled in this thesis is substantially different. As shown in Chapters 4 and 5, the cross-section of a sonar scan with walls and other planar structures results in line-shaped features in the acoustic images. The aspect of these features remains constant independently of the sensor position, although their visibility may change depending on the incidence angle of the emitted beam and many other factors like water turbidity or the structure and materials of the reflecting surface. The resulting acoustic images rarely present shadows since the emitted wave cannot generally pass over the walls and the main part ricochets back to the scenario. Of course, this can cause phantom reflections, especially when operating in confined spaces. The sensorial choice for this work, the MSIS, also presents many differences with respect to the common approach based on electronically scanned sonars. As commented on previously, the MSIS produces scans at a much lower frequency, the data is continuous and is affected by motion-induced distortions. On the other hand,

its visibility is not limited to a reduced scan sector, but can be extended to 360° around the vehicle. This is important because landmarks can be tracked for a longer period of time, being observable even when the vehicle has left them behind.

6.1.1 Classical approaches for line feature extraction

The use of line features has been traditionally related to the use of 2D laser scans in indoor environments. The extensive research carried out in this type of scenario has fostered the development of an abundance of methods for the estimation of lines from a cloud of point measurements. The most popular approaches include algorithms based on segmentation and grouping like Split-and-Merge [Pavlidis, 1982], iterative methods like the RANdom SAmple Consensus (RANSAC) [Fischler and Bolles, 1981] and voting schemes like the Hough transform [Illingworth and Kittler, 1988] among others. It is also worth mentioning the methods based on the Least Squares (LS) minimization, which, although not robust to spurious measurements, is an excellent option to refine lines previously obtained with other methods like Hough or RANSAC.

The approach presented in this thesis relies on the Hough transform, although it has been substantially modified for the application at hand. This choice was motivated by the simplicity of adapting the system to operate with continuous data from the sensor as well as to integrate the necessary motion corrections. Moreover, the Hough transform offers the possibility of detecting other types of features in addition to lines, under the condition that an adequate parametric representation exists [Ballard, 1987]. The particular approach implemented here is based on the work presented in [Tardós et al., 2002]. This work demonstrated a SLAM system running a Hough-based feature extraction algorithm using measurements from a sonar ring mounted on an indoor robot. Although the algorithm proposed here has some similarities, the change in the application domain and the use of an MSIS have motivated important differences with respect to the original.

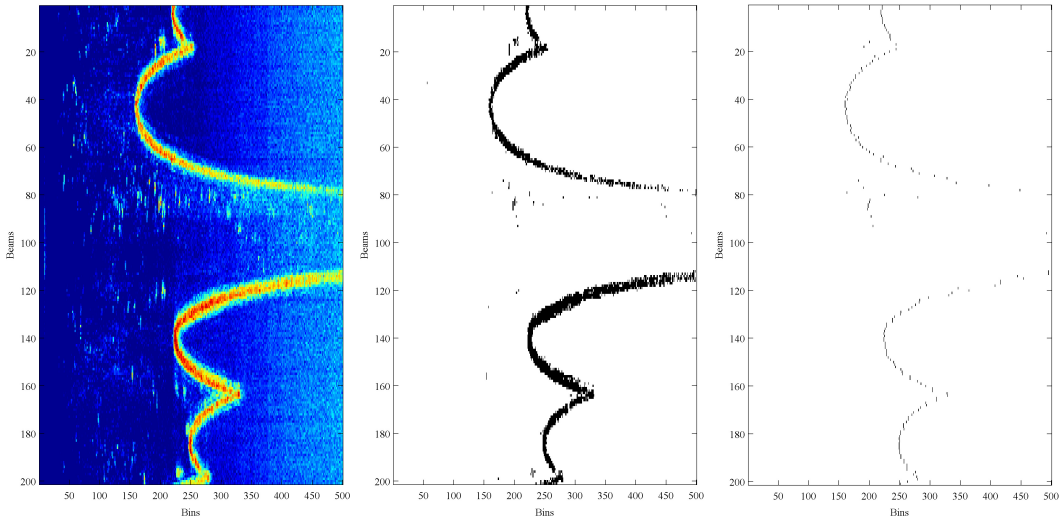
6.1.2 Hough-based feature extraction method for MSIS

This line feature extraction method has some points in common with the voting algorithm for the hybrid localization system presented in Section 5.4. The operation of this feature extraction algorithm is intimately related to the Kalman filter which, simultaneously, executes the SLAM. Not only because it provides the filter with the landmark observations that are necessary to build and mantain the map, but also because the voting algorithm requires position

estimates to deal with the motion induced distorions in the acoustic data. The segmentation of the acoustic data and the use of a buffer to accumulate information are other similarities with the previously introduced localization method. The different aspects of the algorithm will be now described in detail.

Beam segmentation

Objects present in the environment appear as high echo-amplitude returns in acoustic images (see the yellow to red zones in Figure 6.2(a)). Thus, only part of the information stored in each beam is useful for feature extraction. Therefore, a segmentation process can be done in order to obtain the most significant information. This process consists of three steps which are performed beam to beam as they arrive from the sonar. First, only those bins with an intensity value over a threshold are selected and stored. This procedure separates the acoustic imprint left by an object in the image from the noisy background data (Figure 6.2(b)). As will be introduced in Section 6.2, this imprint plays an important role in the estimation of the uncertainty for the features detected with the voting algorithm. The second step is to select from the segmented data above the threshold value, those bins that are local maxima. These high intensity bins are the ones that most likely correspond to objects present in the scene. It is worth noting that this process is performed beam to beam and that as a result of the search for local maxima, one or more high intensity bins can be obtained per beam. The purpose of selecting multiple bins is to make detecting features possible when more than one wall intersects with a single beam. Structures composed of steps or ramps are examples of scenarios where this can happen (see Figure 6.12). Moreover, using this segmentation technique, it is also possible to extract features from ghost reflections. Although it has not been an object of study during this thesis, the invariance and persistence of these reflections suggest that they could be suitable to act as landmarks in a SLAM framework. The last step of the segmentation process is to reject those bins which do not satisfy a "minimum distance between them" criterion. This means that if two bins which have been previously selected as local maxima are too close, they should correspond to the same object and hence, are redundant. Then, the one with the lowest intensity value is discarded (see the resulting high intensity bins in Figure 6.2(c)).



(a) (b) (c)

Figure 6.2: Different phases of the acoustic data segmentation: (a) Raw polar sensor data. (b) Thresholded data. (c) Local maxima bins.

Data buffer

In order to deal with the stream of measurements produced by the continuous arrival of beams, a data buffer similar to the one introduced in Section 5.2.2 is set. This time, the buffer stores information regarding the beams for the most recent 180° scan sector. The choice of this sector size is not arbitrary. The feature extraction algorithm is constantly looking for new lines, which makes detecting a candidate as soon as it is fully represented in the scan possible. Since a 180° sector is the maximum sector that a single line can cover within a scan, there is no need to store more data. Whenever new beams corresponding to an unexplored zone are acquired with the MSIS, the stored information corresponding to old beams that fall outside the most recent 180° scan sector is discarded. When the segmentation process determines that a newly measured beam contains one or more high intensity bins that must take part in the voting, the following information is stored in the buffer:

1. The range and bearing for each of the selected bins (polar coordinates in the sensor reference frame). This information will be used in the voting.

2. The segmented beam obtained after applying the threshold value in the first step of the segmentation process. This will be used during the

uncertainty estimation phase (Section 6.2).

3. The vehicle position estimate at the moment the beam was acquired. This estimate is obtained from the EKF-based SLAM which runs parallel to the feature extraction algorithm. Taking into account these position estimates during the voting compensates the motion-induced distortions in the acoustic data.

Defining the voting space

The information stored in the data buffer is used periodically in a voting to look for possible candidate features. This is performed with a modified version of the classical implementation of the Hough transform for line extraction. This algorithm accumulates the information from the sensor data in a voting table which is a parameterized representation of all the possible feature locations. Those features that receive a great number of votes are the ones with a relevant set of compatible sensor measurements and thus the ones that most likely correspond to a real object in the environment. In our application, line features are described by two parameters, ρ^B and θ^B (perpendicular distance and orientation with respect to a base frame B). Hence, the resulting Hough space is a two-dimensional space where the voting process and the search for maxima can be done efficiently. The base reference frame B can be set arbitrarily. However, our choice for B is the current position of the sonar head at the moment the voting is performed. Since in this implementation the voting is triggered by the arrival of new beams from the sensor, the most recently stored position in the data buffer (the one corresponding to the last beam) defines the position of B. An advantage of choosing this base is that, when a line feature is detected after the voting, its parameters are already represented in the sensor coordinate frame and hence, it can be integrated directly into the SLAM framework as an observation of one of the features already in the map or incorporated as a new feature after compounding it with the current vehicle position. It is worth noting that B is not a fixed coordinate frame. As the parametrization in the Hough space is performed in polar coordinates, setting the reference in a fixed position would produce a resolution loss with an increase in range. To avoid this, we need to re-situate B according to the vehicle's motion. Unfortunately, this requires recomputing the Hough space with each change in the position of B. Although it may seem a great deal of computation, the fact is that the number of bins involved in the voting is not large (less than 100 bins during the tests performed) and the calculations can be executed quite fast. Moreover, as will be explained in the next section, there are situations

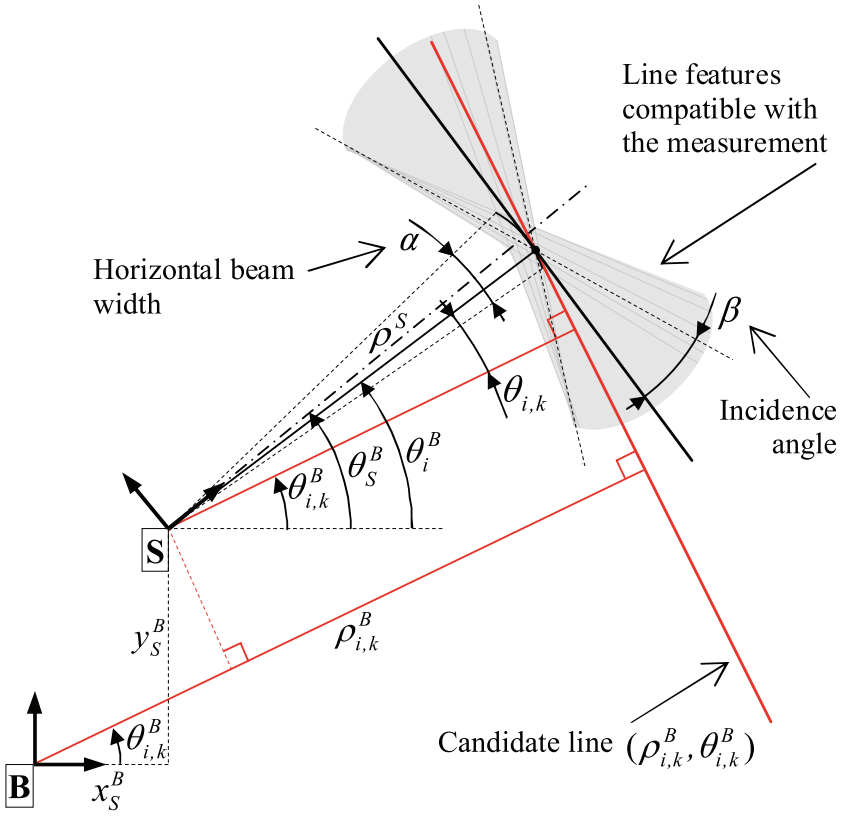


Figure 6.3: Model of the sonar sensor for line features. Where B is the base reference frame and S is a reference frame attached to a beam.

when recalculating the Hough space can be avoided. Another key issue is the quantization of the Hough space. In our case, we have observed that selecting the quantization equal to the angular and linear resolutions of our sensor (typically, 1.8° and 0.1 m) works fine. A higher resolution does not necessarily increase the quality of the detection because the sonar resolution limits its precision. On the other hand, a lower resolution would produce a rough observation.

The general execution of the feature extraction process consists of several steps. First, with each beam arrival, the Hough space is referenced to the current sensor position as the new base frame B. Next, all the bins stored in the buffer are referenced to B so they can be used to vote in the space. It is worth noting that the stored beam positions are taken into account when transforming to B. Hence, the data is undistorted. Then, the votes corresponding to each bin are assigned to the candidate lines by means of a

sonar model. Finally, a search for winning candidates is performed.

Sonar model and voting

Each bin represents the strength of the echo intensity return in a particular place within the insonified area. Due to the uncertainty produced by the horizontal beamwidth, a measurement cannot be assigned to a single point in the space. A common approach [Leonard and Durrant-Whyte, 1992],[Tardós et al., 2002], is to consider the measurement as an arc whose aperture represents the beamwidth uncertainty. Moreover, as a high intensity return is typically produced when the acoustic wave hits a surface perpendicularly, we can infer that all the surfaces tangent to the arc can explain the high intensity return. While this simple model is well suited for air sonar ranging systems, it is not able to explain the acoustic images gathered with an MSIS. A careful analysis of these images reveals that their object detection capability is not limited to the arc-tangent surfaces, but that those beams intersecting the surface within the limits defined by a certain maximum incidence angle also produce a discernible return. On the other hand, those beams with a shallower angle are completely reflected and do not perceive the surface. To obtain a better description of this situation, an extended model to describe the imaging sonar has been adopted (Figure 6.3). Basically, given a horizontal beamwidth angle α (in our sensor, $\alpha = 3°$) and an incidence angle β (generally, not less than $60°$), the set of line features compatible with a particular bin is composed not only of these lines tangent to the arc defined by α, but also of all the lines which intersect the arc with an incidence angle smaller than $\pm\beta$. Before performing a voting, this set of lines must be determined for each bin stored in the data buffer.

This process will now be described using as reference the illustration in Figure 6.3. Let the reference frame S define the position of the transducer head at the moment a particular bin was obtained, with $[x_S^B, y_S^B, \theta_S^B]$ being the transformation which defines the position of S with respect to the chosen base reference B, and ρ^S the range at which the bin was measured from the sensor. Both the transformation and the range values can be obtained from the information in the data buffer. To emulate the effect of the horizontal beamwidth, a set of i values are taken at a given resolution within an aperture of $\pm\alpha/2$ around the direction in which the transducer is oriented, also referred as θ_S^B:

$$\theta_S^B - \frac{\alpha}{2} \leq \theta_i^B \leq \theta_S^B + \frac{\alpha}{2}\,.$$

Each value θ_i^B represents the bearing parameter for a line tangent with the arc which models the horizontal beamwidth. As stated earlier, not only the

lines tangent to the arc are candidates, but also the ones inside the maximum incidence angle limits of $\pm\beta$. For this reason, k values are taken at a given resolution for each value of θ_i^B and within an aperture of $\pm\beta$:

$$\theta_i^B - \beta \leq \theta_{i,k}^B \leq \theta_i^B + \beta\,.$$

The result of this operation are $i \times k$ different values of $\theta_{i,k}^B$. These are the bearings for a set of lines which are a representation of all the possible candidates compatible with the bin. The final step is to determine the range parameter $\rho_{i,k}^B$ corresponding to each one of the $\theta_{i,k}^B$ bearings obtained. Given the geometry of the problem, they are calculated as:

$$\rho_{i,k}^B = x_S^B \cos(\theta_{i,k}^B) + y_S^B \sin(\theta_{i,k}^B) + \rho^S \cos(\theta_{i,k})\,.$$

This set of lines can now be used to determine the cells in the voting space that should receive a single vote from this particular bin. It is assumed that the resolutions chosen during the generation of the $i \times k$ lines are sufficient to ensure a correct exploration of the grid cells and hence, that the zone in the discretized space corresponding to the compatible candidates is correctly determined. This process is repeated for all the bins stored in the data buffer. In Figure 6.4 what the set of voters looks like when assigned to the Hough space is shown. Note that each selected cell of the space can only receive one vote from any particular bin and that those cells containing multiple votes therefore represent lines compatible with different individual bins.

Every time a new beam arrives, a new voting space is generated to look for winning line candidates. A winning line must only be detected once it has been completely observed (i.e., further beams cannot provide more votes to the candidate). In the voting space, the zone in which these winning lines can exist is completely determined by the subset of all the candidate lines contained in the most recent 180° scan sector that do not intersect with the last beam (shaded zones in Fig. 6.4). Any line candidate with a sufficient number of votes found within this zone is declared a winner. Performing the detection in this way can ensure that the algorithm detects the lines as soon as they are completely visible. After a line detection, all the bins involved in the election of the selected candidate are removed from the buffer so they do not interfere with the detection of further features.

It is worth mentioning that, in order to reduce the computational cost of the process, some votings can be skipped. After each voting, it is possible to determine the cell with the largest number of votes and therefore, to calculate the number of supplementary votes required to produce a winner. Since additional votes can only be obtained from newly measured bins, it is

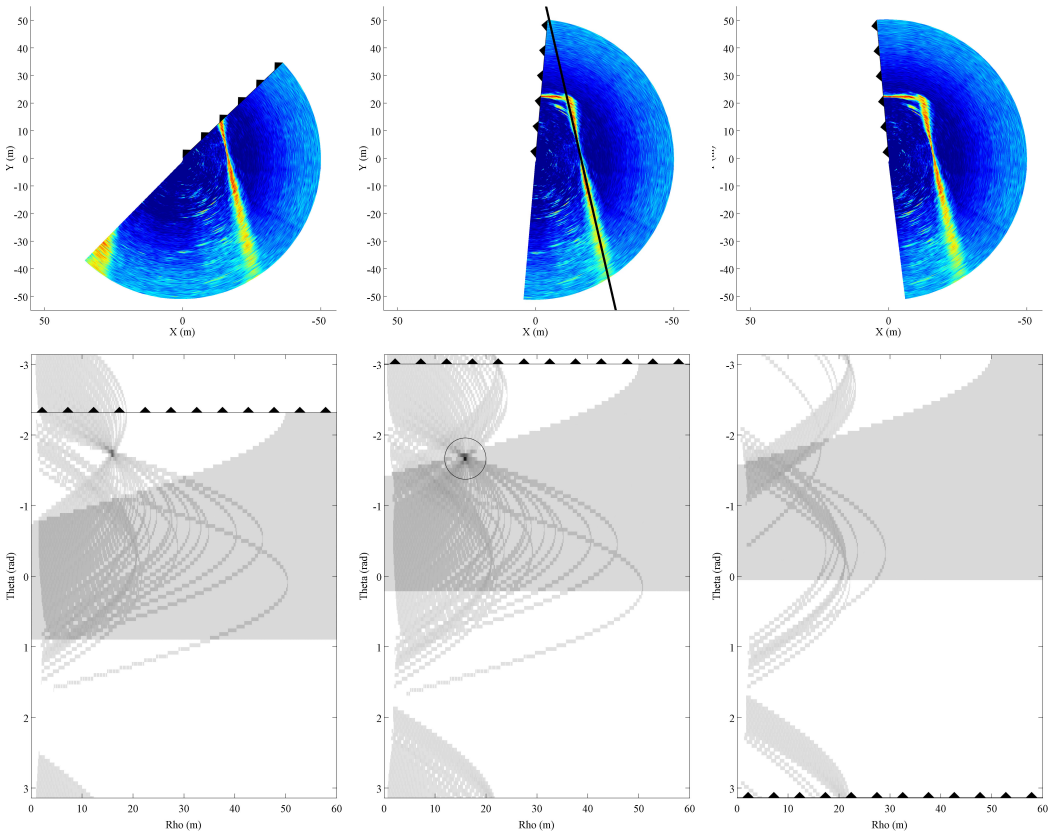


Figure 6.4: Sequence representing the voting process. The scan sector stored in the buffer (top) is represented together with its corresponding voting space (bottom). The line with triangular shapes marks the position of the most recent beam. The darker cells in the voting space represent those candidates with a larger number of votes while the shaded zone represents those candidates which have received all the possible votes. (a) Part of the target line is still outside the sector scan and can receive more votes in the future. (b) The line can now be detected because it has been fully observed and more votes cannot be added. (c) Those votes corresponding to the detected line, as well as the old ones that fall outside the 180° scan sector, are removed from the Hough space so they cannot interfere with future line detections.

not necessary to perform more votings before the minimum required number of bins has been measured and introduced in the buffer.

6.2 Uncertainty model for line features

At this point a method has been introduced to determine the position, in polar coordinates, of a candidate line feature extracted from acoustic data acquired with an MSIS. This information however, is not sufficient to use the line as a landmark for the SLAM system. Although an estimate for the feature parameters has been produced, their values are discrete and their precision depends on the grid resolution of the Hough space. In addition, the line feature lacks an adequate uncertainty model, which makes it impossible integrating the observation in the stochastic map. This section will introduce a novel method for producing an estimate and their uncertainty for line features extracted from rich acoustic data.

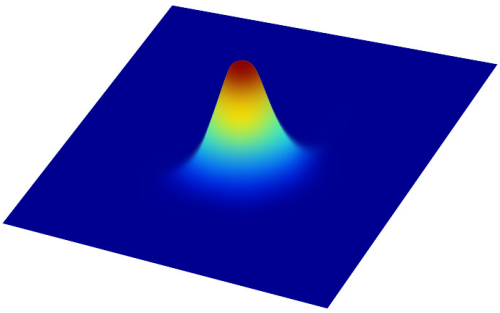
6.2.1 Classical approach

The subset of high intensity bins compatible with the observation of a particular line feature can be determined by analyzing the votes it has received in the Hough space. A common approach to incorporate new features into the stochastic map is to initialize the line feature using the parameters obtained in the Hough space and assigning large covariance values to them (non-informative prior). Then the information from the subset of bins associated with this line feature are used in the form of sensor measurements with the objective of refining the estimate of the feature [Tardós et al., 2002; Ribas et al., 2006]. This method is well suited for laser scans and range measurements in general. However, from the author's point of view, this is not the most adequate method for the estimation of features from the rich data produced by an MSIS since an important part of the information contained in the acoustic images is not taken into account. To optimize the voting in the Hough space, only a small group of selected high intensity bins are used, while neighboring bins with similar intensity values are discarded. Moreover, when a winner is selected, those bins associated with the line candidate are biased into being perfectly aligned, leaving slightly misaligned bins which may also correspond to the real feature outside the estimation process. On the other hand, the noise model assigned to the bins during the estimation of the line is generally based on assumptions regarding the sensor precision, but it ignores the fact that other external factors like water turbidity, incidence angle or characteristics of the reflecting surface, also determine the

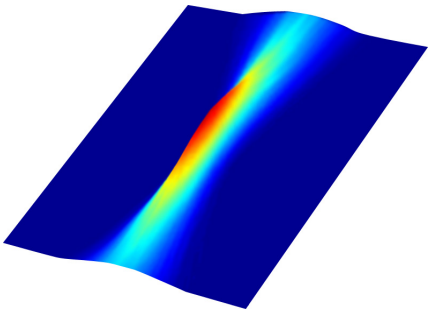
quality of the measurement. For all these reasons, this process usually leads to overoptimistic line estimates which may eventually result in inconsistent maps. A variant of this method seeks to obtain less confident estimations by using only the two bins placed at the endpoints of the line feature instead of using the complete set of measurements. Although the desired effect is obtained, the resulting estimate does not match with the real uncertainty of the feature because of the small number of measurements involved and the inadequacy of the noise model. The method that will be presented in the next section takes a completely different approach. For a human observer analyzing an acoustic image it is fairly simple to discern the zone in which a particular feature will exist with high probability. This zone will not be defined by a few aligned bins but by a large set of bins which look like a blurry elongated shape. The aspect of this shape reflects the quality of the observation, not only depending on the sensor's performance but also on the environment's characteristics. The basic idea for the proposed method is, therefore, focusing on the aspect that features present in acoustic images to determine their uncertainty model rather than relying solely on some particular measurements.

6.2.2 Estimating feature uncertainties from acoustic images

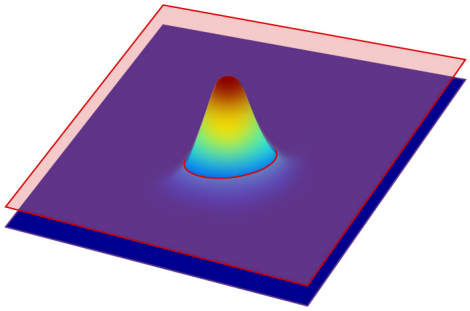
The process to estimate a feature's uncertainty is based on relating the probability of an object existing in a particular place with the measured intensities in the acoustic image representing the same location. There is a high probability that there will be an object in a zone where large intensity values have been measured (e.g. the red-yellow shapes in Figure 6.2(a)) while the probability in the zones with lower intensity measurements gradually decreases to zero (the blue zones in the figure). Given this, the process of applying a threshold to segment the acoustic data can be considered analogous to defining a particular confidence interval for a probability distribution. In other words, a line feature will fall inside the thresholded zone in the acoustic image with a particular confidence level. To make the problem tractable, the probability distribution of a line feature represented in the acoustic image will be approximated to a bivariate Gaussian distribution on its ρ and θ parameters (see Figure 6.5, an additional example justifying that this approximation is suitable can be found in Section 6.2.3). Therefore, the process to estimate the feature uncertainty consists of determining the Gaussian which best fits the segmented data representing a probability distribution for a given confidence level.



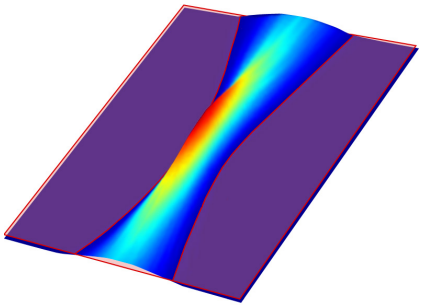
(a) Gaussian distribution representing an uncertain ρ-θ line in a polar space.



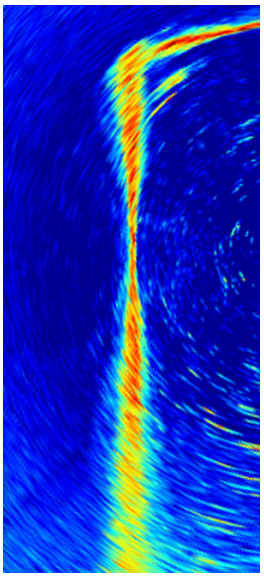
(b) The same uncertain line represented in a cartesian space.



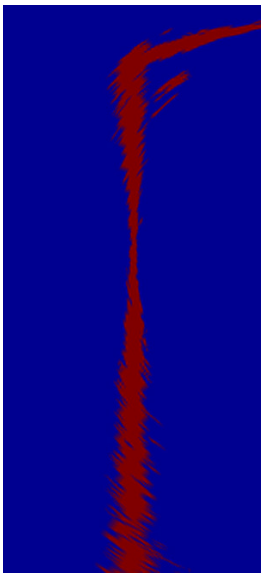
(c) Elliptic section resulting from the definition of a particular confidence interval.



(d) The confidence interval defined in the cartesian representation



(e) The intensities in a cartesian acoustic image are related with the probability of a line feature to exist.



(f) The segmented image resembles the zone defined by the confidence interval in Figure 6.5(d)

Figure 6.5: Relating a segmented acoustic image with a Gaussian probability distribution.

A simple description of this process is shown in Algorithm 1. After the detection of a line feature with the voting algorithm, the uncertainty estimation process begins with the assignment of a feasible confidence coefficient to the imprint left after the segmentation (for instance, it is realistic to assume that the segmented data in Figure 6.2(b) will contain the real feature in the 95% of the cases). Since the winning candidate line has received a considerable number of votes, it must be one of the lines contained within the confidence interval defined by the segmented imprint. The next step of the process consists of finding a number of compatible lines belonging to the neighborhood of the winning candidate which overlap the segmented data in the same way. The objective of this is to obtain a set of line realizations representative of the population contained within the defined confidence interval (i.e. a set of lines that "fill" the segmented area). Estimating the Gaussian distribution from a set of lines is not straightforward, however, it is worth noting that lines described by its ρ and θ parameters can also be represented as points in a polar ρ-θ space. Representing the set of lines in such a space will result in a cloud of points (the lines are similar) with an elliptic form. This particular elliptic disposition of the ρ-θ points suggests that the approximation of the line feature to a Gaussian distribution is correct. Although the space has changed, the set still represents a population of lines within the previously defined confidence interval. This fact is used to estimate the uncertainty of the line feature. It is achieved by approximating the area occupied by the set of points to the area enclosed inside the ellipse that a bivariate Gaussian distribution would generate at the same given confidence. By knowing the confidence coefficient, the major and minor axis of the ellipse and its orientation, it is possible to recover the covariance matrix. Moreover, the mean value of ρ-θ pair defining the line feature can also be obtained from the center of the ellipse.

Figure 6.6 illustrates the different steps involved in the process of estimating the feature uncertainty. The image in Figure 6.6(a) reproduces a voting space which has just obtained a winning candidate (marked with the small box). The corresponding sonar measurements appear in Figure 6.6(b) and are represented in the same B-based polar space that the Hough space. Since the data is represented in polar, the line feature appears as an arc whose thickness is related to its uncertainty. Note that the ρ-θ pair, representing the winning candidate line in the Hough space, can also be represented in this space. In fact, to parametrize the line, its point with the smallest distance to the origin is used (again, represented with the same small box in the figure). Applying a threshold and assigning a confidence coefficient to

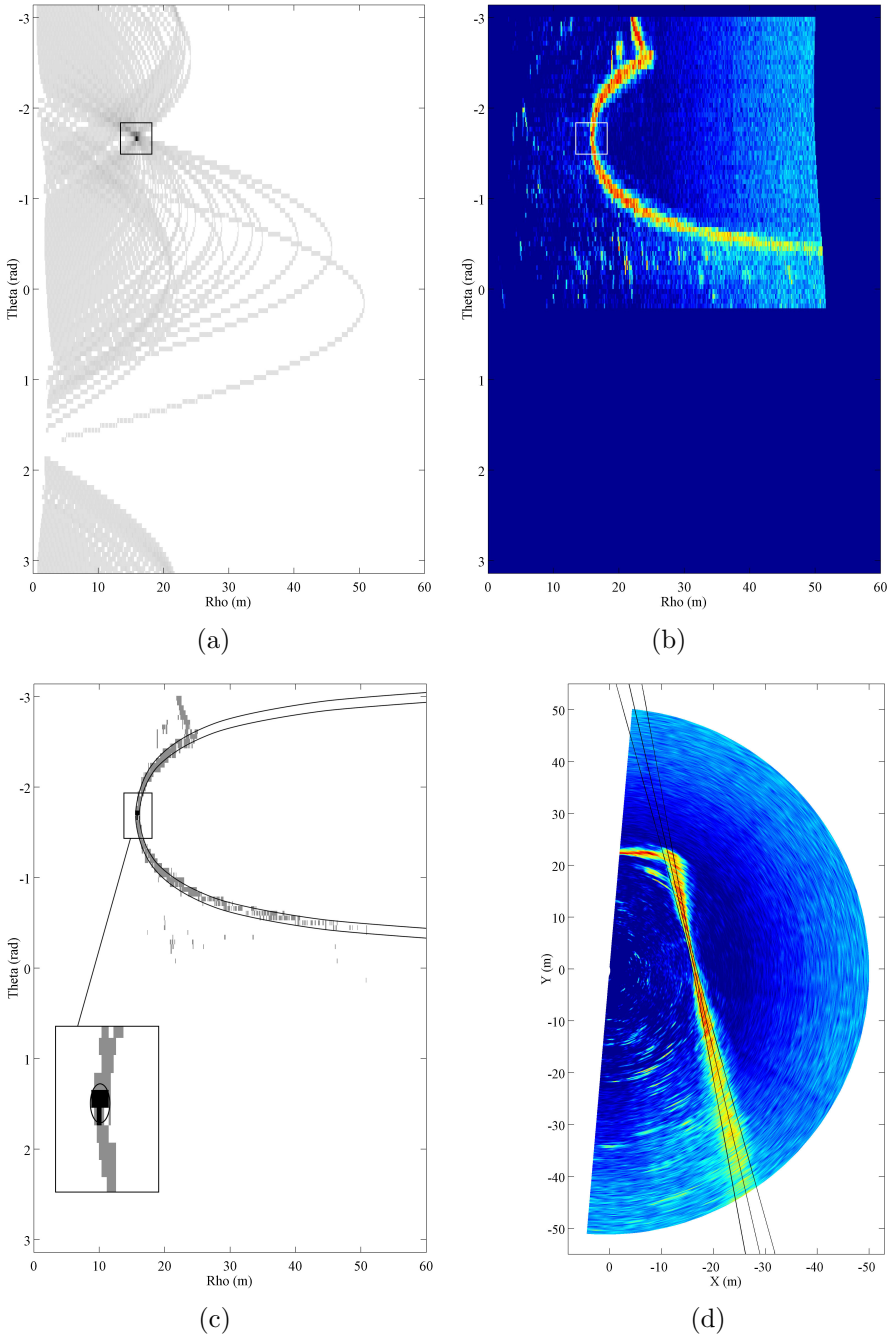


Figure 6.6: Process for uncertainty estimation. (a) Winning candidate in the Hough space. (b) Polar representation of the sonar data. (c) Segmented data with the zone occupied by line features inside the confidence level. (d) Resulting uncertainty estimate represented over the scan sector.

Algorithm 1: $get\_measurements([\rho_c, \theta_c], \text{scan}, \text{confidence\_level})$

```
/* Initialization of the polar grid space that will
   contain the segmented sonar data                          */
boolean last180scan [ρ_resolution, θ_resolution];
[last180scan] = init_scan(scan);

/* Set the paradigm with the candidate line from the
   voting                                                    */
[η_c] = get_overlap_ratio([ρ_c, θ_c], last180scan);

/* Search for compatible lines                               */
lines2check = {[ρ_c, θ_c]};
accepted = ∅;
rejected = ∅;
while lines2check ≠ ∅ do
    [ρ_i, θ_i] = get_candidate(lines2check);
    [η_i] = get_overlap_ratio([ρ_i, θ_i], last180scan);
    if accept_line(η_c, η_i) then
        accepted = accepted ∪ {[ρ_i, θ_i]};
        lines2check = lines2check\{[ρ_i, θ_i]};
        lines2check = lines2check ∪ {neighbour8connectivity([ρ_i, θ_i]) ∩
        {rejected ⋃ accepted}})
    else
        rejected = rejected ∪ {[ρ_i, θ_i]};
        lines2check = lines2check\{[ρ_i, θ_i]};

/* Given the set of lines, determine the ellipse that
   contains the area where they exist                        */
[major_axis, minor_axis, ρ_mean, θ_mean, α] = get_ellipse(accepted);

/* Given the ellipse and the confidence level related to
   the segmentation, find the mean and covariance            */
z^V = [ρ_mean, θ_mean];
R = get_covariance(major_axis, minor_axis, ρ_mean, θ_mean, conf_level)

return [z^V, R];
```

the segmented data results in the space represented in Figure 6.6(c). At this point, and using the winning candidate line as a paradigm, the search for lines contained within the segmented imprint is performed. The resulting set of lines is contained inside the bounds represented as black arcs, while the representation of the place occupied by their ρ and θ pairs is represented as a black shape at the apex of the arc. The final step of the procedure consists of finding the ellipse containing this area and extracting the covariance matrix given the predefined confidence coefficient. Finally, Figure 6.6(d) represents the estimated feature over a cartesian representation of the scan sector. The line in the center corresponds to the ρ-θ mean value while the lines at the sides represent the uncertainty bounds at a 95% of confidence.

6.2.3 Validation of the feature extraction algorithm

In order to validate the feature extraction algorithm, several tests with both synthetic and real data have been carried out. With generating synthetic data we seek two objectives. The fist one is to justify the use of a bivariate ρ-θ Gaussian distribution to represent the uncertain features present in the acoustic images. The second is to have a way of comparing the output from the algorithm with the paradigm which makes it possible to confirm the correctness of the estimation. To obtain the synthetic data, a large population of ρ-θ pairs was generated following a given probability distribution. Then, the lines represented by each pair were projected into a polar space analogous to those produced by the measurements from an MSIS. Each cell from this space represents a bin and its echo intensity value is assigned accordingly to the number of lines that cross its area. The resulting synthetic dataset is represented in polar and cartesian coordinates in Figures 6.7(a) and 6.7(d). In spite of the large uncertainty assigned with the goal of improving the visualization of the uncertainty estimation process, the example has sufficient points in common with the real acoustic images to consider this model as valid. It can be observed how the high intensity zone in the center corresponds with the major concentration of lines, while the dispersion on the sides, caused by the angular uncertainty, produces an effect similar to the loss of intensity and precision affecting the beams with large incidence angles. Figures 6.7(b) and 6.7(c) illustrate the voting and the uncertainty estimation process. The elliptic shaped zone representing the population of compatible lines reflects the Gaussianity of the estimated feature. As can be observed in Figure 6.7(d), the estimated line feature is a good representation of what appears in the synthetic data.

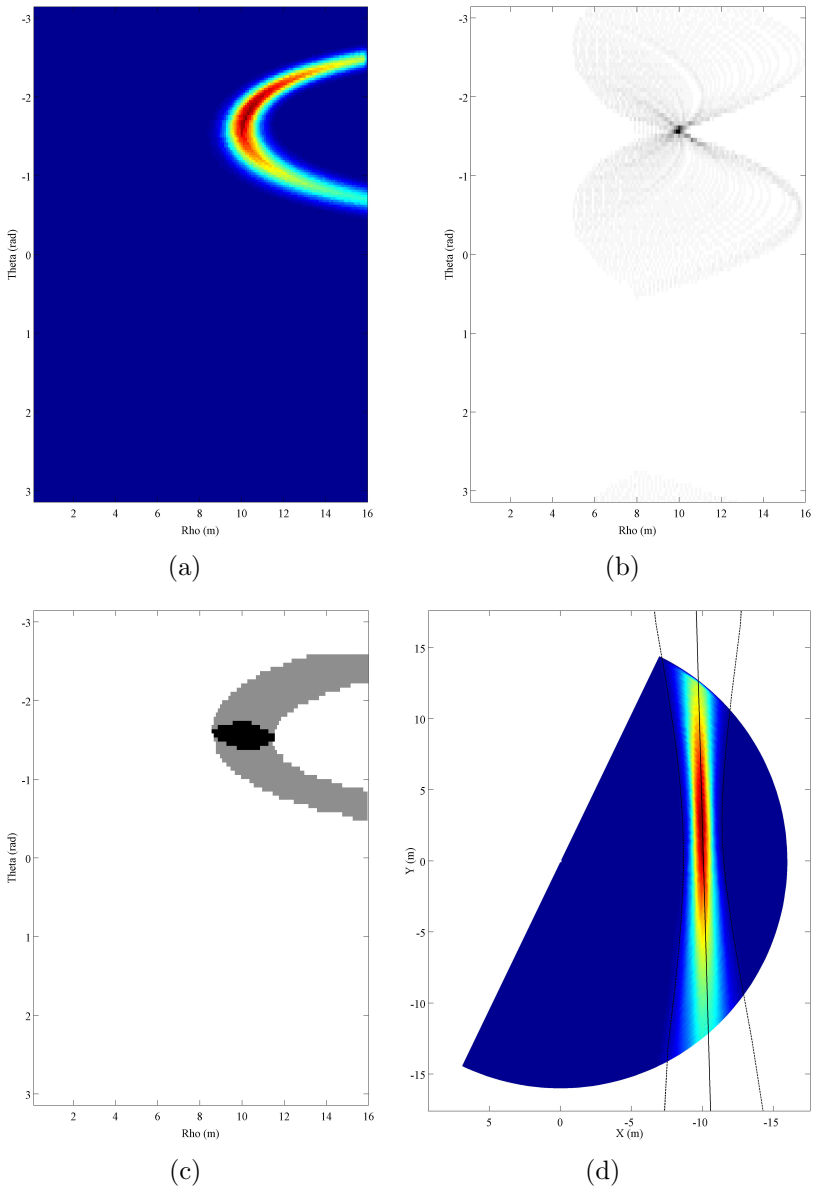


Figure 6.7: Testing the algorithm with synthetic data. (a) Raw sensor data generated from ρ and θ given a normally distributed uncertainty. Some correlation affects the two variables to increase the difficulty of the test. (b) The voting space clearly identifies the line. (c) Uncertainty estimation using the segmented data. The black elliptic shape corresponds to the lines with compatible overlapping and represents the uncertainty of ρ and θ. (e) The estimated line feature fits almost perfectly with the synthetic one.

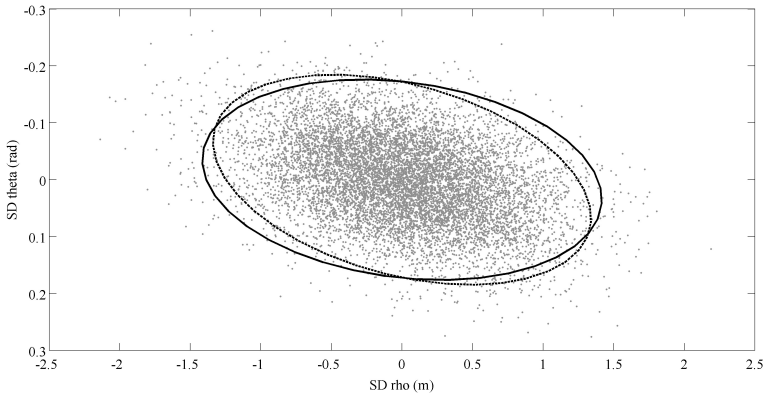


Figure 6.8: Comparison between the bivariate Gaussian distribution used to produce the synthetic data and the output from the algorithm. The ellipse with dashed line represents the Gaussian distribution at a 95% confidence while the solid one is the output of algorithm at the same confidence level.

Additional verification of the method can be seen in Figure 6.8, where the cloud of ρ-θ pairs initially used to generate the synthetic data is plotted together with an ellipse representing the original Gaussian distribution (dashed line) and another one representing the one estimated with the proposed method (solid line). When comparing the two ellipses, it can be appreciated that they are almost coincident except for a small angular misalignement. It is important to note that correlated data, like the one in this example, has turned out to be the most difficult scenario for the proposed uncertainty estimation method and therefore, one could expect even better estimates when working with less correlated data.

A second set of tests was carried out with real data acquired with the Ictineu AUV. Under real working conditions, it is not possible to obtain reliable references to test the performance of the method. Therefore, only the direct visualization of the estimated line feature represented over the acoustic images can be used as an indicator. The first example in Figure 6.9(a) shows the features extracted from a dataset obtained in a real application scenario; in particular, in the same marina environment which served as the testbed for the SLAM algorithm. The second example is represented in Figure 6.9(b). It corresponds to an experiment performed in the water tank of the Underwater Robotics Research Center at the University of Girona. This confined environment with highly reflective concrete walls produces noisy data with many reflections and phantoms. The results in both cases are consistent with the representation of the walls in the acoustic images and, moreover, it shows reliable behavior when working with noisy data, filtering

linear features from shapeless phantoms.

6.3 Obtaining segments

The last step in the process of acquiring features from acoustic images is to determine the line segments. This method takes advantage of the same segmented data used during the uncertainty estimation process as well as the mean value of the estimated line parameters. The process consists basically of determining the overlap of the estimated line over the thresholded data and finding the line segments which are placed over high intensity areas. Then a process equivalent to the *dilation* and *erosion* morphological operations for image processing is applied to the obtained segments in order to group and remove short segments and to produce a more compact representation. Finally, the resulting segment endpoints are referenced as distances, measured along the line, from the point represented by the ρ-θ parameters (the point in which the line is closer to the coordinate reference frame) and stored for their posterior use. It is worth noting that these segments are produced for representation purposes only. The segment endpoints are not introduced in the state vector and hence, they are not estimated in any way.

6.4 EKF-based SLAM

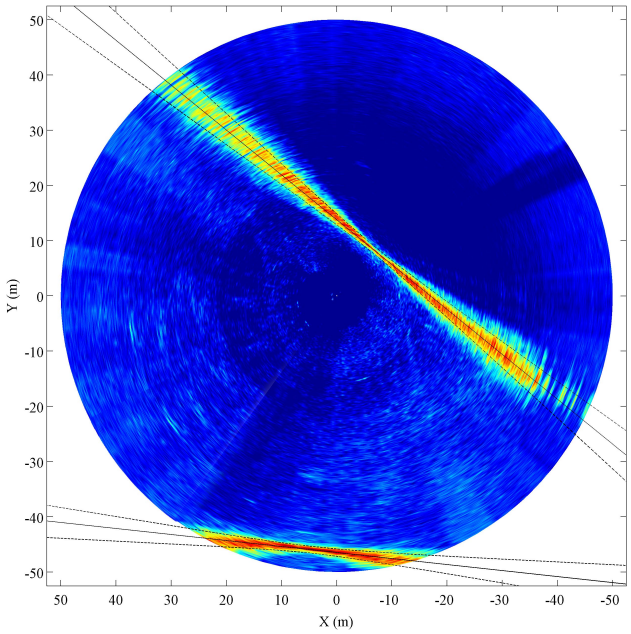
An EKF integrates the vehicle's navigation sensors to provide an estimate of its position and retain the estimates of the previously observed features in order to build a map. This filter is an implementation of the stochastic map [Smith et al., 1990] in which the estimate of the position of both the vehicle $\mathbf{x}_V$ and the set of map features $F = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$ are stored in the state vector $\hat{\mathbf{x}}$.

$$\hat{\mathbf{x}}(k) = [\hat{\mathbf{x}}_V(k) \quad \hat{\mathbf{x}}_1(k) \ \dots \ \hat{\mathbf{x}}_n(k)]^T$$

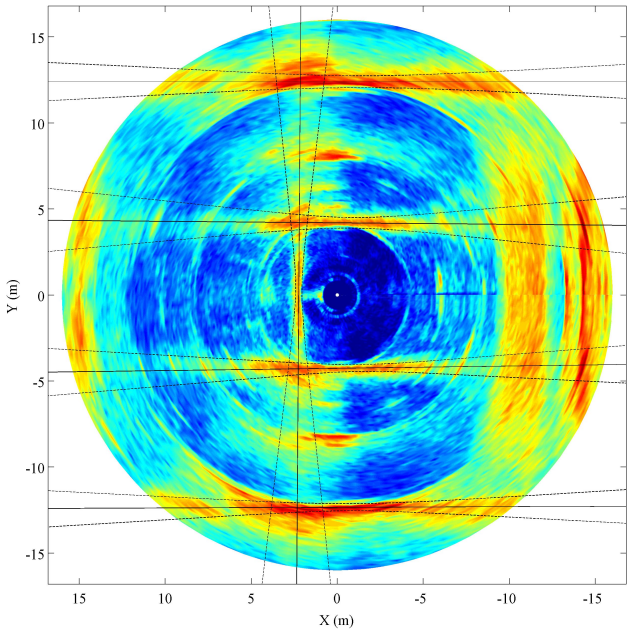
The covariance matrix $\mathbf{P}$ describes the covariance of the vehicle and the features as well as their respective cross correlations:

$$\mathbf{P}(k) = E([\mathbf{x}(k) - \hat{\mathbf{x}}(k)][\mathbf{x}(k) - \hat{\mathbf{x}}(k)]^T | Z(k))$$

The vehicle's state $\hat{\mathbf{x}}_V$ has dimension 9, which defines the minimum size of the state vector $\hat{\mathbf{x}}$ at the beginning of the execution. The features are represented in polar coordinates and therefore, the state will be increased by 2 with each new incorporation in the map.



(a)



(b)

Figure 6.9: Testing the algorithm with real data. (a) Line features extracted from acoustic data gathered in a marina environment. (b) Line features obtained from acoustic data gathered in a small water tank. The lines on the right side are not estimated as they are split between the start and the end of the scan.

6.4.1 Map initialization

When creating a new stochastic map at step 0, a base local reference frame L must be selected (Fig. 6.10). In this approach, the initial vehicle position is chosen to set this base location and thus, is initialized with perfect knowledge. The vehicle's state $\mathbf{x}_V$ is represented as:

$$\mathbf{x}_V = [x\ y\ z\ \psi\ u\ v\ w\ r\ \psi_{L_0}]^T$$

where $[x\ y\ z\ \psi]$ represent the position and heading of the vehicle in the local reference frame L while $[u\ v\ w\ r]$ are their corresponding linear and angular velocities on the vehicle's coordinate frame V. As can be seen, the vehicle's state vector is exactly the same as the one presented in Section 5.3, except for the term ψ_{L_0}, which represents the angle between the initial vehicle heading at step 0 (orientation of L) and magnetic north in the earth global frame E. This term works as a sensor bias and allows us to initialize the vehicle heading ψ in the local frame L , making it possible to use compass measurements (angle to the north in the E frame) for its estimation as shown in Section 6.4.3. Assuming that the vehicle is not moving at step 0, the state is initialized as:

$$\hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_V(0) = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \hat{\psi}_{L_0}\right]^T; \quad \mathbf{P}(0) = \mathbf{P}_V(0) = \begin{bmatrix} \mathbf{0}_{8\times 8} & \mathbf{0}_{8\times 1} \\ \mathbf{0}_{1\times 8} & \sigma^2_{\psi_{L_0}} \end{bmatrix}$$

where $\hat{\psi}_{L_0}$ takes its value from the first available compass measurement and $\sigma^2_{\psi_{L_0}}$ is initialized accordingly with the sensor's precision. It is worth noting that at the beginning of the execution, the map does not contain any feature and hence, the state $\hat{\mathbf{x}}$ contains only the vehicle's state $\hat{\mathbf{x}}_V$.

6.4.2 Prediction

Again, the previously introduced constant velocity kinematics model is used to predict the state of the vehicle (see Section 5.3.4). The new term ψ_{L_0} has been added and modeled as constant since the original orientation of the local map does not changes with time:

$$\mathbf{x}_V(k) \ = \ f(\mathbf{x}_V(k-1), \mathbf{n}(k-1)),$$

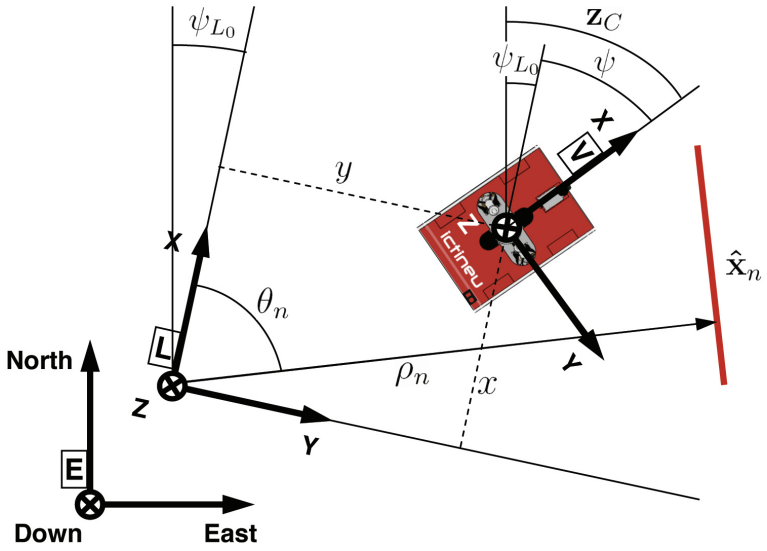


Figure 6.10: Representation of the different reference coordinate frames.

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \\ \psi_{L_0} \end{bmatrix}_{(k)} = \begin{bmatrix} x+(uT+n_u\frac{T^2}{2})cos(\psi)-(vT+n_v\frac{T^2}{2})sin(\psi) \\ y+(uT+n_u\frac{T^2}{2})sin(\psi)+(vT+n_v\frac{T^2}{2})cos(\psi) \\ z+wT+n_w\frac{T^2}{2} \\ \psi+rT+n_r\frac{T^2}{2} \\ u+n_uT \\ v+n_vT \\ w+n_wT \\ r+n_rT \\ \psi_{L_0} \end{bmatrix}_{(k-1)} .$$

The noise model is also the same, with $\mathbf{n} = [n_u \; n_v \; n_w \; n_r]^T$ representing an acceleration white noise additive in the velocity terms which has a zero mean and covariance $\mathbf{Q}$. On the other hand, as features correspond to fixed objects from the environment, we can assume they are stationary. Hence, the whole state can be predicted as:

$$\hat{\mathbf{x}}(k|k-1) = [f(\hat{\mathbf{x}}_V(k-1)) \quad \hat{\mathbf{x}}_1(k-1) \; \ldots \; \hat{\mathbf{x}}_n(k-1)]^T$$

and its covariance matrix updated as:

$$\mathbf{P}(k|k-1) = \begin{bmatrix} \mathbf{F}_V(k) & \mathbf{0}_{9\times 2n} \\ \mathbf{0}_{2n\times 9} & \mathbf{I}_{2n\times 2n} \end{bmatrix} \mathbf{P}(k-1) \begin{bmatrix} \mathbf{F}_V(k) & \mathbf{0}_{9\times 2n} \\ \mathbf{0}_{2n\times 9} & \mathbf{I}_{2n\times 2n} \end{bmatrix}^T + \begin{bmatrix} \mathbf{W}_V(k) \\ \mathbf{0}_{2n\times 4} \end{bmatrix} \mathbf{Q} \begin{bmatrix} \mathbf{W}_V(k) \\ \mathbf{0}_{2n\times 4} \end{bmatrix}^T$$

where $\mathbf{F}_V$ and $\mathbf{W}_V$ are the Jacobian matrices of partial derivatives of the non-linear model function f with respect to the state $\mathbf{x}_V$ and the noise $\mathbf{n}$

respectively:

$$\mathbf{F}_V = \frac{\partial f}{\partial \mathbf{x}_V}(\hat{\mathbf{x}}_V(k), \mathbf{0}) = \begin{bmatrix} 1 & 0 & 0 & -\hat{u}T\sin\hat{\psi} - \hat{v}T\cos\hat{\psi} & T\cos\hat{\psi} & -T\sin\hat{\psi} & 0 & 0 & 0 \\ 0 & 1 & 0 & \hat{u}T\cos\hat{\psi} - \hat{v}T\sin\hat{\psi} & T\sin\hat{\psi} & T\cos\hat{\psi} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{W}_V = \frac{\partial f}{\partial \mathbf{n}}(\hat{\mathbf{x}}_V(k), \mathbf{0}) = \begin{bmatrix} \frac{T^2}{2}\cos\hat{\psi} & \frac{-T^2}{2}\sin\hat{\psi} & 0 & 0 \\ \frac{T^2}{2}\sin\hat{\psi} & \frac{T^2}{2}\cos\hat{\psi} & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & 0 & \frac{T^2}{2} \\ T & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

6.4.3 Sensor updates

The measurements from the DVL, the pressure sensor and the compass are treated as described in Section 5.3.5. However, the measurement model equations should be adapted to deal with the changes introduced in the state vector:

Velocity: The velocity measurements provided by the DVL are integrated as direct observations of the vehicle's velocities in the state. The observation matrix $\mathbf{H}$ is adapted to the dimension of the state vector as:

$$\mathbf{H}_D = \begin{bmatrix} \mathbf{0}_{3\times 4} & \mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 2} & \mathbf{0}_{3\times 2n} \end{bmatrix},$$

while the covariance matrix for the measurement noise remains as:

$$\mathbf{R}_D = \begin{bmatrix} \sigma_{Du}^2 & 0 & 0 \\ 0 & \sigma_{Dv}^2 & 0 \\ 0 & 0 & \sigma_{Dw}^2 \end{bmatrix}.$$

Depth: The pressure sensor measurements are also integrated as a direct observation of the vehicle's depth (position in the Z axis). The adapted **H** matrix is:

$$\mathbf{H}_P = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0}_{1\times 2n} \end{bmatrix}.$$

The variance of the depth measurement will be represented by:

$$\mathbf{R}_P = \sigma_P^2.$$

Heading: Working in a local map makes modifying the measurement model to update the vehicle's heading necessary. As can be observed in Fig. 6.10, the compass measurement $\mathbf{z}_C$ corresponds to the addition of the heading of the vehicle ψ with respect to the local reference frame L and the orientation of this frame ψ_{L_0}. The resulting measurement model is:

$$\mathbf{H}_C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & \mathbf{0}_{1\times 2n} \end{bmatrix}.$$

The measurement noise $\mathbf{R}_C$ is represented by the variance of the compass:

$$\mathbf{R}_C = \sigma_C^2.$$

Again, the measurements from the different sensors will be integrated as soon as they are acquired using the classical Kalman filter update equations for linear measurement models. In order to deal with simultaneously arriving measurements, a reconfigurable composed form of the **H** matrix equivalent to the one presented in Section 5.3.5 is used. The same is applicable in the case of the use of bottom and/or water tracking velocities from the DVL.

6.4.4 About the use of a compass in SLAM

Working with compass data can be a difficult task in some situations. The effect of electromagnetic fields, like those produced by the thrusters, and the presence of large structures with ferromagnetic materials can considerably distort its measurements and render them unusable. Nowadays, there are alternative technologies like Fiber Optic Gyro (FOG) and Ring Laser Gyro (RLG) which offer perturbation-free high precision measurements [Kinsey et al., 2006]. However, these devices are very expensive and their use is not viable for low-cost vehicles. Although it is almost impossible to completely avoid the effect of perturbations on compasses, taking certain precautions such as performing calibrations before each mission and avoiding operating

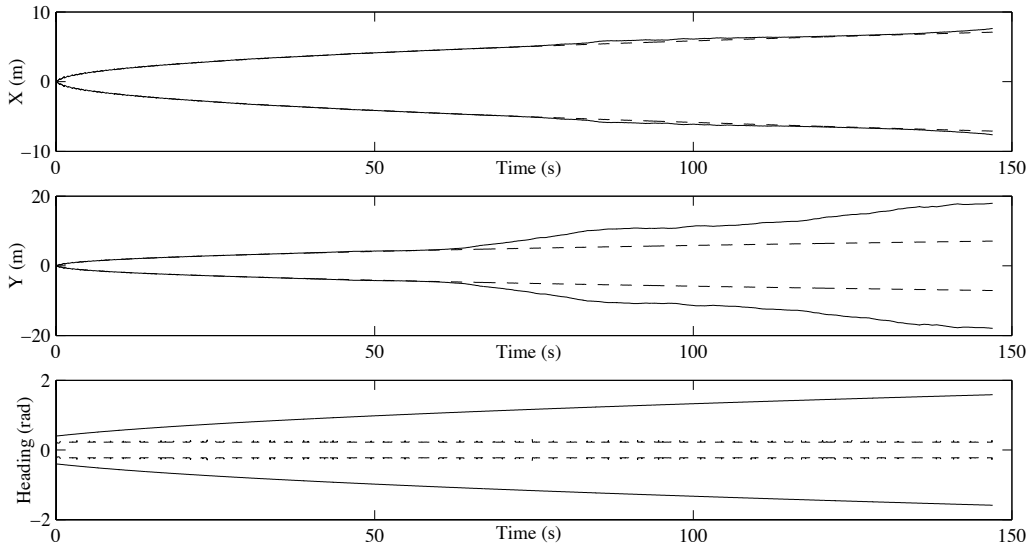


Figure 6.11: Estimated position covariance plots represented within 2σ bounds. This data corresponds to the first minutes of the abandoned marina experiment executing the EKF with the updates from the dead-reckoning sensors. The results using a cheap inaccurate gyro sensor are represented with a solid line, while the ones using absolute data from a compass are indicated with a dashed line.

close to walls (generally, 1-2 m is sufficient), can provide measurements of sufficient quality. Moreover, a compass is an especially useful sensor for SLAM because it provides *absolute* orientation measurements, unlike the dead reckoning sensors normally used in SLAM such as wheel encoders, gyros or, in our case, the DVL. The effect of using a compass is threefold:

1. The error in vehicle orientation will *not* increase during the SLAM process.

2. Vehicle orientation introduces nonlinearity in the SLAM problem, so loss of precision because of linearization effects will also be limited.

3. Vehicle orientation errors in a certain step become position errors in future steps. Bounding the errors in orientation will also result in a reduction in the rate of increase of vehicle position errors.

Figure 6.11 shows the evolution of vehicle's position and orientation using the DVL velocity data together with the rate of turn measurements from gyros (solid line) and, using *absolute* attitude information from the compass

(dashed line). We can see that the error in orientation remains constant. There is also a reduction in the rate of increase of the error in the direction transverse to the vehicle's direction of motion.

6.4.5 Map building process

The Tritech Miniking imaging sonar produces beams at a 10-30Hz rate depending on the settings of the sensor. Each new beam is stored together with the current vehicle position estimate from the filter in the data buffer and fed to the feature extraction algorithm as shown in Section 6.1. Eventually, the information added by a new beam arrival is sufficient to produce a line feature detection. In this case, the ρ - θ pair obtained is represented in the B frame which is placed in the current position of the sonar head. For the sake of simplicity, let us assume that the transformation between B and the vehicle's coordinate system is known. Hence, we could represent a new measurement i with respect to the vehicle's frame V as $\mathbf{z}_i^V = [\rho_i^V \ \theta_i^V]^T$. Of course, the same transformation should be applied to the covariance matrix obtained from the uncertainty estimation method. This transformation will result in the covariance matrix $\mathbf{R}_i$. The next step is to solve the data association problem. That is to determine if the measured line $\mathbf{z}_i^V$ corresponds to any of the features $F_j, j = 1 \dots n$ already existing in the map and should be used to update the system or, on the contrary, it is new and has to be incorporated into the map. The result of the data association process is a hypothesis $\mathcal{H} = \mathtt{j}_i$ associating the measurement $\mathbf{z}_i^V$ with one of the map features F_j ($\mathtt{j}_i = 0$ indicates that $\mathbf{z}_i^V$ has no correspondence with the existing features). Finding the correct hypothesis is a process involving the analysis of the discrepancy between the actual line measurement and its prediction. This prediction is obtained from the nonlinear measurement function h_j, which relates the i measurement with the state vector $\mathbf{x}(k)$ containing the locations of the vehicle and the j feature:

$$\begin{aligned} \mathbf{z}_i^V(k) &= h_j(\mathbf{x}(k), \mathbf{s}_i), \\ \begin{bmatrix} \rho_i^V \\ \theta_i^V \end{bmatrix} &= \begin{bmatrix} \rho_j - x\cos\theta_j - y\sin\theta_j \\ \theta_j - \psi \end{bmatrix} + \mathbf{s}_i, \end{aligned}$$

where $\mathbf{s}_i$, the noise affecting the line feature observation, is a zero-mean white noise with covariance $\mathbf{R}_i$. To calculate the discrepancy between the measurement and its prediction, the innovation term ν_{ij} and its associate covariance matrix $\mathbf{S}_{ij}$ are obtained as:

$$\boldsymbol{\nu}_{ij}(k) = \mathbf{z}_i^V(k) - h_j(\hat{\mathbf{x}}(k|k-1)),$$

$$\mathbf{S}_{ij}(k) \;=\; \mathbf{H}_j(k)\mathbf{P}(k|k-1)\mathbf{H}(k)_j^T + \mathbf{R}_i,$$

where $\mathbf{H}_j$ represents the Jacobian matrix which linearizes the nonlinear measurement function h_j around the best available estimation of the state $\hat{\mathbf{x}}(k|k-1)$:

$$\begin{aligned}\mathbf{H}_j &= \frac{\partial h_j}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k|k-1), \mathbf{0}) \\ &= \begin{bmatrix} -\cos\theta_j & -\sin\theta_j & 0 & 0 & 0\,0\,0\,0\,0 & \cdots & 1 & x\sin\theta_j - y\cos\theta_j & \cdots\, 0\;0 \\ 0 & 0 & 0 & -1 & 0\,0\,0\,0\,0 & \cdots & 0 & 1 & \cdots\, 0\;0 \end{bmatrix}.\end{aligned}$$

To determine if the correspondence is valid, an individual compatibility (IC) test using the Mahalanobis distance is carried out:

$$D_{ij}^2 = \boldsymbol{\nu}_{ij}(k)^T \mathbf{S}_{ij}(k)^{-1} \boldsymbol{\nu}_{ij}(k) < \chi_{d,\alpha}^2,$$

where $d = \dim(\mathbf{h}_j)$ and α is the desired confidence level. It is possible for multiple hypothesis relating the measurement with different map features to satisfy the IC test. Then, in order to select the best candidate, the nearest neighbor (NN) criterion is applied (in situations where clutter or vehicle uncertainty are high, more complex data association algorithms such as JCBB [Neira and Tardós, 2001] can be used). Finally, after the correspondence has been decided, it is used to update the state estimate by means of the EKF update equations.

$$\begin{aligned}\mathbf{K}_{ij}(k) &= \mathbf{P}(k|k-1)\mathbf{H}_j(k)^T \mathbf{S}_{ij}(k)^{-1}, \\ \hat{\mathbf{x}}(k) &= \hat{\mathbf{x}}(k|k-1) + \mathbf{K}_{ij}(k)\boldsymbol{\nu}_{ij}(k), \\ \mathbf{P}(k) &= (\mathbf{I} - \mathbf{K}_{ij}(k)\mathbf{H}_j(k))\mathbf{P}(k).\end{aligned}$$

In case there is no valid hypothesis relating the measured line with any of the features from the map (i.e. $\mathcal{H} = 0$), this measurement can be added to the current state vector as a new feature. However, this cannot be done directly because this new feature needs to be represented in the map reference frame. The change of reference is done by compounding (see Appendix B) the line feature with the current vehicle position as follows:

$$\hat{\mathbf{x}}(k) = \begin{bmatrix} \hat{\mathbf{x}}_V(k) \\ \hat{\mathbf{x}}_1(k) \\ \vdots \\ \hat{\mathbf{x}}_n(k) \end{bmatrix} \quad \Rightarrow \quad \hat{\mathbf{x}}(k)^+ = \begin{bmatrix} \hat{\mathbf{x}}_V(k) \\ \hat{\mathbf{x}}_1(k) \\ \vdots \\ \hat{\mathbf{x}}_n(k) \\ \hat{\mathbf{x}}_V(k) \oplus \mathbf{z}_i^V(k) \end{bmatrix}.$$

Augmenting the state vector also requires updating the estimated error covariance matrix as:

$$\mathbf{P}(k) = \mathbf{D}(k)\mathbf{P}(k)\mathbf{D}(k)^T + \mathbf{G}(k)\mathbf{R}_i\mathbf{G}(k)^T,$$

$$\mathbf{D}(k) = \begin{bmatrix} \mathbf{I} & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \mathbf{I} \\ \mathbf{J}_{1\oplus} & 0 & \dots & 0 \end{bmatrix}, \quad \mathbf{G}(k) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{J}_{2\oplus} \end{bmatrix},$$

where $\mathbf{J}_{1\oplus}$ and $\mathbf{J}_{2\oplus}$ are the Jacobian matrices of the compounding transformation.

6.5 SLAM with local maps

In recent years, many different authors have proposed methods to carry out SLAM by building sequences of local maps [Leonard and Feder, 2001; Tardós et al., 2002; Williams et al., 2002; Leonard and Newman, 2003; Bosse et al., 2003; Newman et al., 2003; Bosse et al., 2004; Estrada et al., 2005; Clemente et al., 2007; Ni et al., 2007]. The main advantages of building sequences of local maps are the limitation of the cost associated with the update of a full covariance matrix [Guivant and Nebot, 2001] and the improvement of the system's consistency [Castellanos et al., 2004; Huang and Dissanayake, 2007]. In the present case, an additional advantage is obtained with using local maps. The parametrization of line features using polar coordinates is the most adequate approach for our type of sensor (polar). However, it is not the best choice for referencing the features in a large map. Some issues appear when an observation of a new feature is translated from the sensor frame to the map base frame, particularly in those situations where the map base and the sensor base are far from each other, since a small variation in the θ parameter of a feature with a large ρ value translates in large changes in Cartesian coordinates. Using local maps overcomes this issue as their area is smaller and hence, the reference changes are less critical.

An important restriction of such methods is that the local maps must be statistically independent (no information can be shared between them) to avoid introducing inconsistency when recovering the global map. As a consequence, vehicle states such as velocities or estimated sensor biases cannot be transferred between maps. Recently, [Piniés and Tardós, 2007] presented a technique which overcomes this limitation and makes sharing information between local maps possible, while remaining conditionally independent. This is especially useful in our case because it allows to keep information about

the vehicle's state. This method has been chosen to implement the local map sequencing in the present work. Although this section summarizes the main characteristics of our particular implementation of the algorithm, a more detailed presentation of the method can be found in the bibliographic reference mentioned.

6.5.1 Local map building

The local map building process relies on defining a set of state variables which are common to two consecutive maps. This commonality serves as a link to transmit the information from one map to the other while maintaining their conditional independence. In the application at hand, this link makes it possible to use the estimates of the vehicle's velocities and the compass bias obtained at the end of a map to initialize the next local map. Moreover, after new measurements modify the estimate of these terms, it is also possible to update their estimated values in the previous map through back-propagation.

The procedure to build the local maps begins by initializing the filter presented in Section 6.4. Then, the vehicle moves through the scenario acquiring sensor information regarding its own state and the position of existing features. After a certain time period, the state vector $\hat{\mathbf{x}}$ will contain the current estimate of the states of the vehicle $\hat{\mathbf{x}}_V$ as well as the position of several map features $F = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$. At a given instant k, the current local map is finished and a new one is initialized by defining a new state $\hat{\mathbf{x}}$ containing only the current vehicle state $\hat{\mathbf{x}}_V$ as follows:

$$\hat{\mathbf{x}}(k) = \left[\hat{\mathbf{x}}_V(k) \quad \mathbf{T}\hat{\mathbf{x}}_V(k)\right]^T,$$

where the first term is a clone of the vehicle's state that will serve as a link between the two local maps, and the second term represents the initialization of the vehicle's state in the new map after performing a change of the base reference defined by the linear transformation function $\mathbf{T}$:

$$\begin{aligned} \mathbf{T}\hat{\mathbf{x}}_V(k) &= \left[0 \;\; 0 \;\; 0 \;\; 0 \;\; u \;\; v \;\; w \;\; r \;\; \psi{+}\psi_{L_0}\right]^T, \\ \mathbf{T} &= \begin{bmatrix} \mathbf{0}_{4\times4} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times1} \\ \mathbf{0}_{4\times4} & \mathbf{I}_{4\times4} & \mathbf{0}_{4\times1} \\ 0\;0\;0\;1 & \mathbf{0}_{1\times4} & 1 \end{bmatrix}. \end{aligned}$$

This transformation sets the current vehicle location as the base reference of the new local map, while its velocity estimates (represented by the vehicle's frame) are preserved. It is important to note that the term of the compass bias is also updated to make integrating compass measurements with respect to the new base possible. The resulting state vector has a dimension of 18.

To complete the initialization process, the state covariance matrix $\mathbf{P}$ has to be set accordingly:

$$\mathbf{P}(k) = \begin{bmatrix} \mathbf{P}_V(k) & \mathbf{P}_V(k)\mathbf{T}^T \\ \mathbf{T}\mathbf{P}_V(k) & \mathbf{T}\mathbf{P}_V(k)\mathbf{T}^T \end{bmatrix},$$

where $\mathbf{P}_V$ is the submatix corresponding to the vehicle's state from the full covariance matrix of the first map. At this point, the filter is ready to begin the estimation of the new local map using the equations presented in Section 6.4. Of course, those equations should be adapted to the presence of the common state variables representing the link between the maps.

6.5.2 Local map Joining

The map building procedure will result in a sequence of local maps with the form:

$$\mathcal{M}_i = (\hat{\mathbf{x}}^i, \mathbf{P}^i); \quad \text{with} \quad \hat{\mathbf{x}}^i = \begin{bmatrix} \hat{\mathbf{x}}_V^{i-1} & \hat{\mathbf{x}}_1^i \dots \hat{\mathbf{x}}_n^i & \hat{\mathbf{x}}_V^i \end{bmatrix}^T. \tag{6.1}$$

Each local map $\mathcal{M}_i$ contains the term $\hat{\mathbf{x}}_V^{i-1}$, a copy of the vehicle's state at the end of the previous map $\mathcal{M}_{i-1}$ which represents the common part connecting the two maps. It also contains a set of features $\{\hat{\mathbf{x}}_1^i \dots \hat{\mathbf{x}}_n^i\}$ which have been added to the state vector during the generation of the map and, finally, the term $\hat{\mathbf{x}}_V^i$, which represents the estimate of the vehicle's state throughout the creation of the map and whose final value will serve to initialize the $\mathcal{M}_{i+1}$ local map.

The process of joining local maps into a single global map is described here using a notation similar to the one presented in [Piniés and Tardós, 2007]. Consider, two consecutive local maps defined as:

$$\begin{aligned} \mathcal{M}_A &= \left(\begin{bmatrix} \hat{\mathbf{x}}_A \\ \hat{\mathbf{x}}_{Ca} \end{bmatrix}, \begin{bmatrix} \mathbf{P}_A & \mathbf{P}_{ACa} \\ \mathbf{P}_{CaA} & \mathbf{P}_{Ca} \end{bmatrix} \right), \\ \mathcal{M}_B &= \left(\begin{bmatrix} \hat{\mathbf{x}}_{Cb} \\ \hat{\mathbf{x}}_B \end{bmatrix}, \begin{bmatrix} \mathbf{P}_{Cb} & \mathbf{P}_{CbB} \\ \mathbf{P}_{BCb} & \mathbf{P}_B \end{bmatrix} \right). \end{aligned}$$

The part common to both maps is represented by $\hat{\mathbf{x}}_{Ca}$, which corresponds to the state of the vehicle at the end of $\mathcal{M}_A$, and $\hat{\mathbf{x}}_{Cb}$, which is initialized as an exact clone of $\hat{x}_{Ca}$ during the creation of the $\mathcal{M}_B$ map but evolves because of the updates propagated through the correlation terms during the generation of $\mathcal{M}_B$. The rest of the information stored in the maps is represented by $\hat{\mathbf{x}}_A$ and $\hat{\mathbf{x}}_B$. According to the general form described in (6.1), $\hat{\mathbf{x}}_A$ will contain the common term representing the link with a previous map and all the features

in $\mathcal{M}_A$, while $\hat{\mathbf{x}}_B$ will contain the features in $\mathcal{M}_B$ and the estimate of the vehicle's state at the end of the map.

The objective of the map joining process is to obtain a single global map containing the information from all the local maps. In this example, the global map is represented by:

$$\mathcal{M}_{AB} = \left(\begin{bmatrix} \hat{\mathbf{x}}'_A \\ \hat{\mathbf{x}}_{Cb} \\ \hat{\mathbf{x}}_B \end{bmatrix}, \begin{bmatrix} \mathbf{P}'_A & \mathbf{P}'_{ACb} & \mathbf{P}'_{AB} \\ \mathbf{P}'_{CbA} & \mathbf{P}_{Cb} & \mathbf{P}_{CbB} \\ \mathbf{P}'_{BA} & \mathbf{P}_{BCb} & \mathbf{P}_B \end{bmatrix} \right)$$

The last two blocks of the global map coincide exactly with $\mathcal{M}_B$ (they are up to date). Therefore, only the terms related to $\mathbf{x}_A$ need to be updated (a tilde is used to denote those terms). This is because the first map has only been updated with its own measurements but does not contain any information obtained during the generation of the second map. In order to transmit the effect of those measurements to the estimates in the $\mathcal{M}_A$ map, a back-propagation procedure is carried out:

$$\begin{aligned} \mathbf{K} &= \mathbf{P}_{ACa}\mathbf{P}_{Ca}^{-1} \\ \mathbf{P}'_{ACb} &= \mathbf{K}\mathbf{P}_{Cb} \\ \mathbf{P}'_A &= \mathbf{P}_A + \mathbf{K}(\mathbf{P}'_{CbA} - \mathbf{P}_{CaA}) \\ \hat{\mathbf{x}}'_A &= \hat{\mathbf{x}}_A + \mathbf{K}(\hat{\mathbf{x}}_{Cb} - \hat{\mathbf{x}}_{Ca}) \end{aligned}$$

Moreover, in order to recover the full covariance matrix of the global map, it is necessary to calculate the correlation term relating the two local maps:

$$\begin{aligned} \mathbf{P}'_{AB} &= \mathbf{P}'_{ACb}\mathbf{P}_{Cb}^{-1}\mathbf{P}_{CbB} \\ &= \mathbf{K}\mathbf{P}_{CbB} \end{aligned}$$

At this point, all the elements in $\mathcal{M}_{AB}$ have been determined. It is important to note that this map joining procedure is applicable to sequences of more than two local maps. After each union, the resulting map still contains the common elements that serve as a link with the adjacent ones therefore, the same procedure can be applied.

Each element from the resulting global map is still represented in the base frame of their respective local maps. Moreover, it is possible that some features could have been observed from different local maps and therefore they are repeated. The final part of this procedure consists of transforming all the features to a common coordinate frame (see the operators described in Appendix B). Data association can be carried out and, after obtaining the correspondences, the global map can be updated to produce a better estimate. In the context of this work, the Joint Compatibility Branch and Bound

(JCBB) data association algorithm has been used [Neira and Tardós, 2001] to obtain the hypothesis relating features from different local maps. Then, an implicit measurement equation representing the equivalence between paired features is used to perform the update [Castellanos and Tardós, 1999].

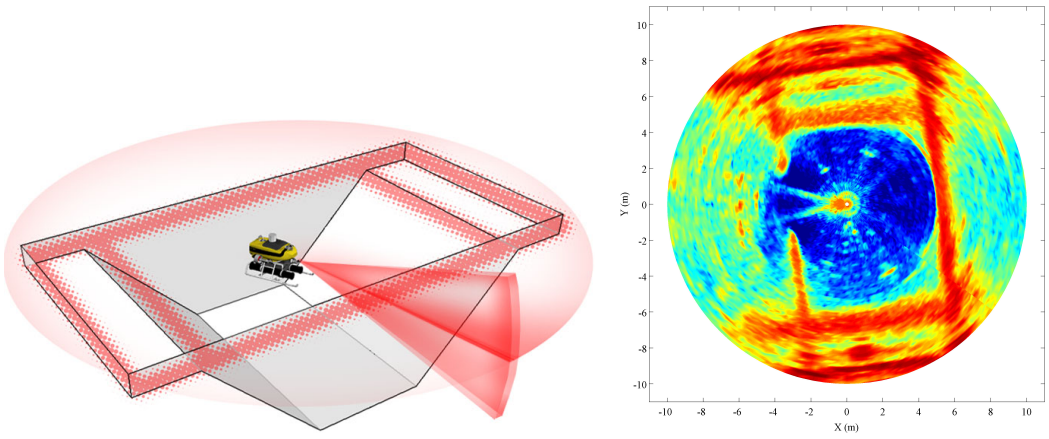
6.6 Experimental results

This section presents two experiments used to test the proposed SLAM approach. The first corresponds to the same CIRS water tank test previously used in Chapter 5 to test various localization algorithms, while the second is an experiment performed in an abandoned marina environment.

6.6.1 Water tank

The CIRS water tank is a hard scenario in which to perform SLAM. The reduced dimensions and the reflectivity of the surfaces make it difficult for the DVL to operate correctly and, as a consequence, the sensor produced an important number of erroneous velocity measurements. As mentioned in Section 5.3.7, both bottom and water velocities are integrated in order to improve the estimate of the vehicle's motion. The compass is also sensible to the presence of nearby structures. Although the vehicle was operated at a distance from the walls (about 1 meter) it is not possible to ensure that the heading measurements are free of perturbations. However, the pessimistic uncertainty model for the vehicle's heading seem to cope with this problem. The operation of the MSIS is not simple, either. The confined space produces noisy measurements and, although the range of the sensor was set to avoid phantom reflections from the boundary walls, there are still some sonar artifacts affecting the data. Also, the tank has a particular geometry with two inclined planes (ramps) at each side. As can be seen in Figure 6.12, those planes appear as wide stripes in the acoustic image. The width of these zones is related with the slant angle of the surfaces. The more slanted the ramps are, the wider the stripes become. Moreover, this increase in the inclination is usually associated to a decrease in the measured intensity, especially when compared with the values produced by a vertical wall. Another characteristic of these ramps is that the position of the stripes will change with the depth of the vehicle, making them an unreliable landmark for a SLAM system. In this test however, the vehicle moved at a constant depth and the inclination of the ramps produced sufficiently narrow stripes. For these reasons, they have been used as landmarks. It is worth noting that, if necessary, these types of features can be discarded by either selecting a higher threshold value or

setting a larger value for the "minimum distance between them" criterion that filters high intensity bins in the sonar data segmentation process (see Section 6.1.2).



(a) (b)

Figure 6.12: Acquisition of MSIS data in the CIRS water tank. (a) Schematic representation of the scenario. The highlighted zones correspond to the places where high intensity bins are expected. (b) The resulting acoustic image of the tank.

Figure 6.13 represents the resulting trajectory (red line) and the six line features composing the map (black). Because of the reduced dimensions of the test scenario, it was not necessary to create of local maps. The dimensions match those of the real water tank (8 by 16 meters) and, as can be seen, the features corresponding to the ramps are correctly placed at each side of the tank, symmetrically and at equal distance from the center. It is important to note that line segments are longer than the actual dimensions of the walls. This is mainly because of the sonar reflections that may occur at the corners of the water tank, which extend the high intensity zones beyond the limits of the walls (see the example in Figure 6.12(b)). It was not possible to establish a ground truth for the vehicle's location during the experiment. However, we can compare the output of the SLAM algorithm with the trajectories estimated with the different localization methods presented in Chapter 5. This is shown in Figure 6.14. As expected, the resulting trajectory is very similar to the reference ones and, again, the approach avoids the drift caused by the use of dead-reckoning sensors.

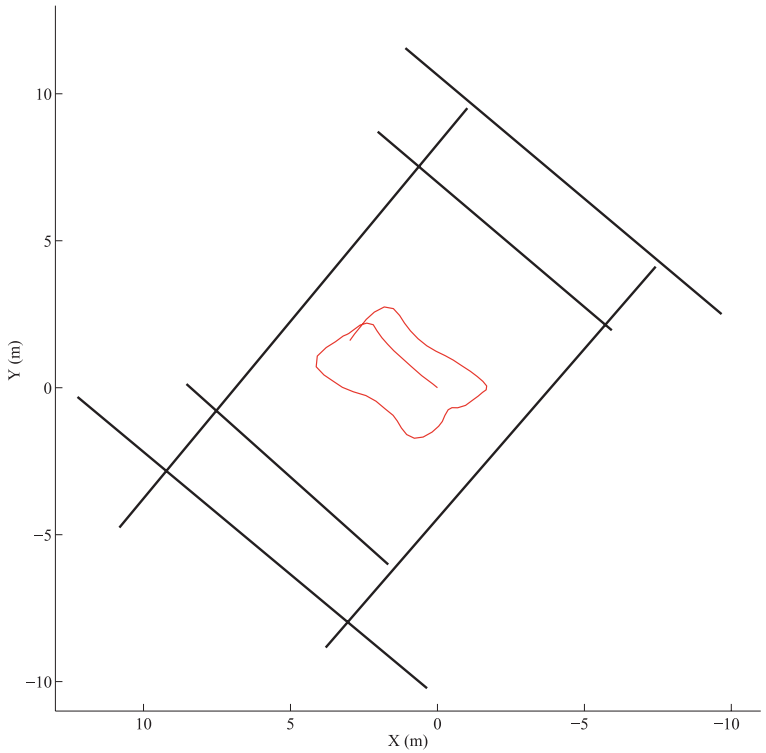


Figure 6.13: The SLAM trajectory (red) and the resulting map (black) for the CIRS water tank dataset.

6.6.2 Marina Environment

In order to test the reliability of the proposed algorithm in a real application scenario, an experiment was carried out in an abandoned marina situated near St. Pere Pescador on the Costa Brava (Spain) [Google Maps, 2008]. The Ictineu AUV gathered a data set along a 600 m operated trajectory which included a small loop around the larger water area and a 200 m straight path through an outgoing canal (see Figure 6.15). The vehicle moved at about 0.2 m/s and the experiment lasted 50 min. The data set included measurements from the DVL, the compass and the imaging sonar, which was set to a range of 50 m, with a resolution of 0.1 m and 1.8°. For validation purposes, the vehicle was operated close to the surface attached to a GPS equipped buoy used for registering the trajectory. In the test scenario, only the boundary walls contain ferromagnetic elements, while the bottom is natural. The vehicle always moved at a considerable distance from these walls and hence, we can assume that there is little distortion affecting the compass. However, this cannot be confirmed since no ground truth is available for the angle

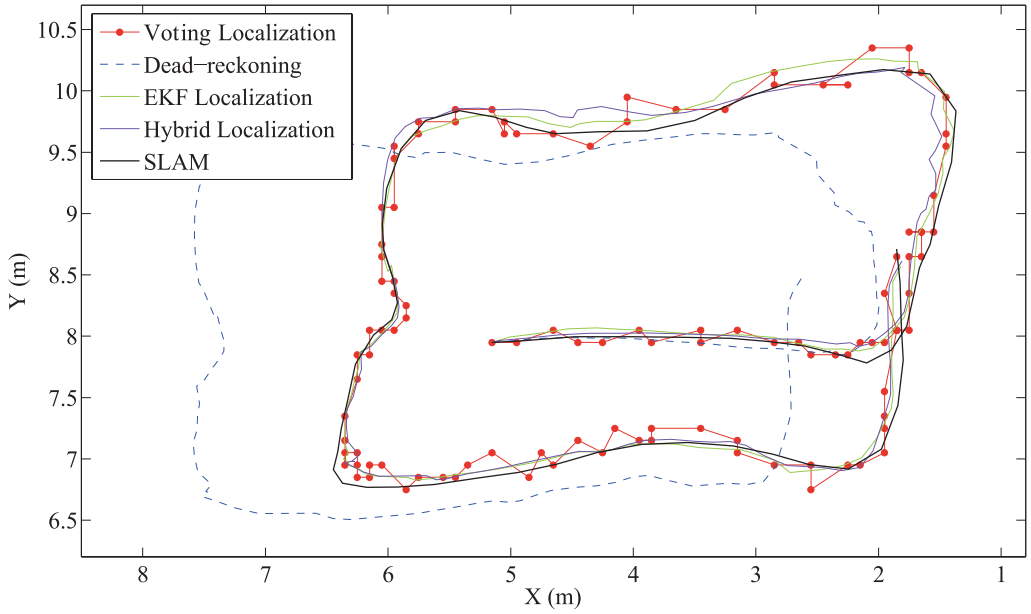
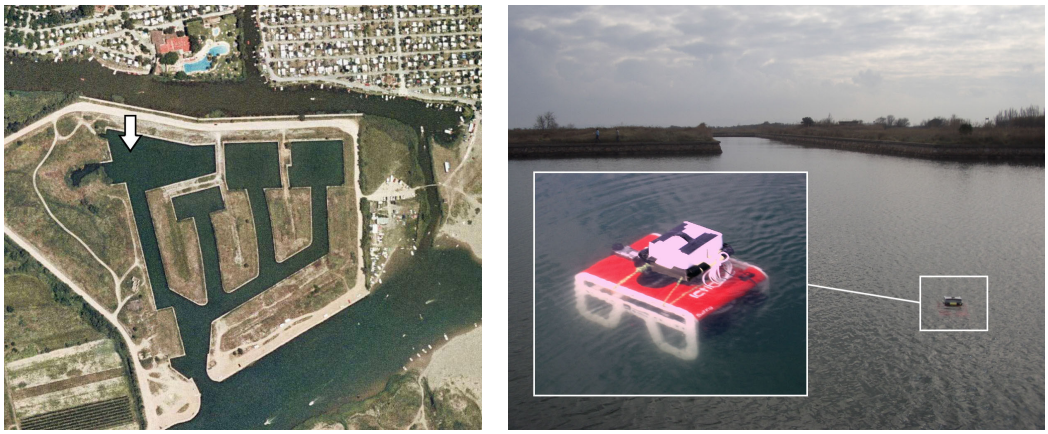


Figure 6.14: Comparison between the SLAM trajectory and the ones obtained with the presented localization methods .

measurements. On the other hand, the DVL measurements were much more reliable than those obtained in the water tank. Only the 3% of the bottom tracking velocities received a bad status indicator from the sensor. Therefore, it was not necessary to rely on water velocities during the execution. The configuration parameters of the SLAM algorithm for this dataset are the same as those used in the water tank test. Only the threshold value for the segmentation of the sonar data has been adapted because the gain setting of the MSIS was different in the two experiments. In the water tank, the gain was too high, which resulted in more saturated images, while during the current experiment, it was set to a more adequate value.

Figure 6.16 represents the trajectory obtained during the generation of the different submaps (solid black line), which is a good approximation to the one measured with the GPS (dashed). As commented on before, one of the benefits of working with small local maps is that it improves the behavior of line features represented in polar coordinates. For this reason, a distance from the origin of 75 meters is set as the condition to initialize a new local map (the limits are represented by circles in the figure). It is worth noting the sudden position change that appears in the estimated trajectory at approximately [-40, 25]. This correction is a consequence of re-observing, in the second local map and after performing a small loop, the features at the beginning of the long canal. Given the shape and dimensions of the



(a) (b)

Figure 6.15: Experiment in the abandoned marina. (a) Ortophotomap of the test area. (b) The Ictineu AUV equipped with a surface buoy during the experiment.

scenario and the range setting of the sonar, the few places where a loop closure could happen are limited to the principal area. The path followed towards the top part of this area is split between the two first local maps. Therefore, the only place where a loop closure could happen is in the lower part of the area, when the features at each side go out of sight. In these loop-closing situations, a discontinuity is introduced in the trajectory stored in the sonar data buffer. It is, however, uncommon for such strong position corrections to affect the feature extraction process. The update that produces this discontinuity generally takes place just after the complete observation of a feature and during the initial moments of the next one. Therefore, the major part of the new bins introduced into the buffer will usually be obtained on the already corrected track. It can also be observed how the discrepancy with the GPS data increments when the vehicle moves through the canal. This is mainly caused mainly by the absence of features placed perpendicularly to the direction of the canal axis, which makes the correction of the errors accumulating in this direction difficult.

The global map and the estimated trajectory (solid line) obtained after the joining are plotted in Figure 6.17 layered over a satellite image. For comparison, the GPS trajectory (dashed line) and a dead-reckoning trajectory (dot-dashed line), obtained by executing the filter with only the measurements from the DVL and the compass, are also represented. As can be observed, the dead-reckoning data suffers from an appreciable drift (even

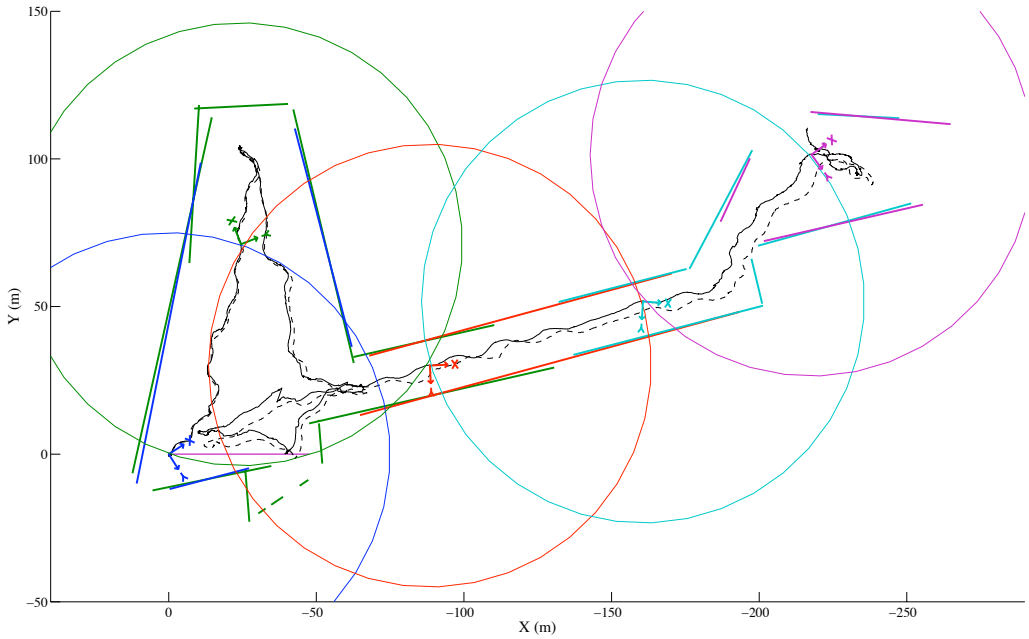


Figure 6.16: Sequence of local maps. The SLAM trajectory is represented with a solid line and the DGPS with a dashed line. Different colors represent each one of the local maps, their boundaries and base frames.

causing it to go outside the canal), while the SLAM estimated trajectory follows the GPS track with considerable precision. The resulting map is also a good approximation, matching almost perfectly with the real position of the marina's boundaries. A problem with the position of a feature is observed in the upper-left part of the map. This effect is due to the similarity between the two intersecting lines. The small intersection angle makes it difficult for the feature extraction to discern between the two lines and, eventually, they are interpreted as a single (slightly distorted) one. Of course, this also affects the measurement of the segment endpoints, as it makes it difficult to determine the overlapping with the thresholded data and tends to make longer segments. Some minor problems with the measurement of the segment endpoints are also observed in the small channel entrance in the lower-left part of the map. They mainly appear because of the polar parametrization used in the line features, which, in some particular situations, produces a misplacement of the segment endpoints.

Figure 6.18 represents the error plots for the resulting estimated trajectory obtained after producing the local map. The GPS data has been used as the ground truth. As can be seen, the error is contained within the 2σ limits, confirming the correct operation of the SLAM.

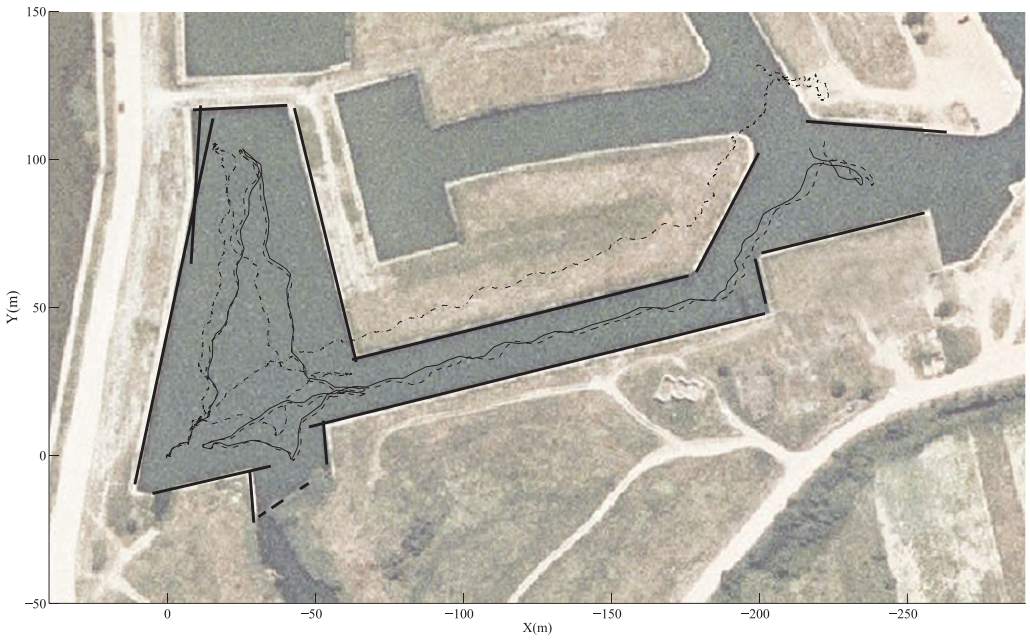


Figure 6.17: The resulting global map together with the dead-reckoning (dash-dotted line), GPS (dashed line) and SLAM (solid line) trajectories represented over a satellite image of the scenario.

Additional results validating the algorithm are shown in Figure 6.19, which reproduces two acoustic images generated by placing the sonar measurements from the complete dataset according to the dead-reckoning and the SLAM estimated trajectories. An averaged representation of all the overlapping scans have been used; therefore, one can expect the diffuse appearance shown on the dead-reckoning image as a result of the dispersion induced by the erroneous trajectory. On the other hand, using the SLAM trajectory provides a more accurate placement of the measurements which results in a sharper image.

Only the acquisition of the sensor data was performed in real time by the computers onboard the Ictineu. This SLAM approach was implemented on Matlab and executed off-line on an ordinary desktop computer. The execution time is smaller than the duration of the real experiment. Therefore, it is not unrealistic to assume that a more optimized implementation should be able to operate onboard.

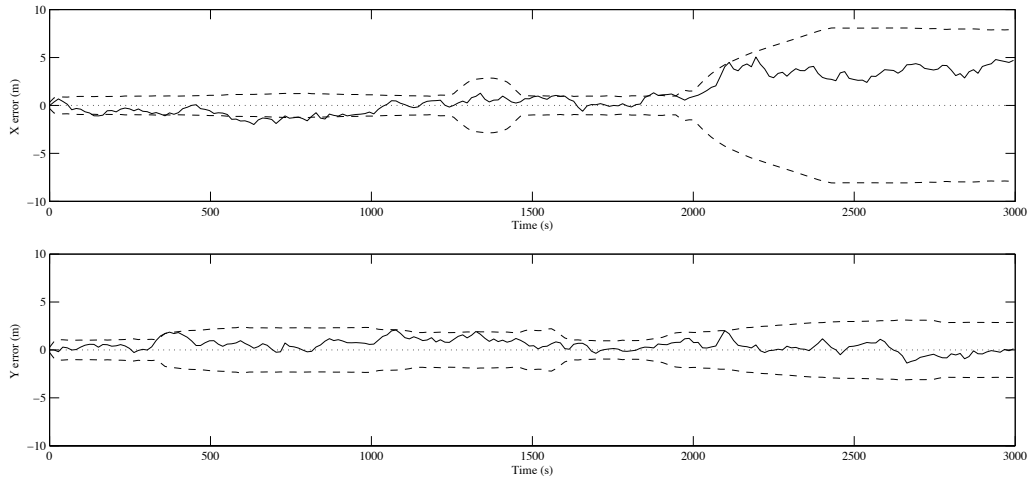
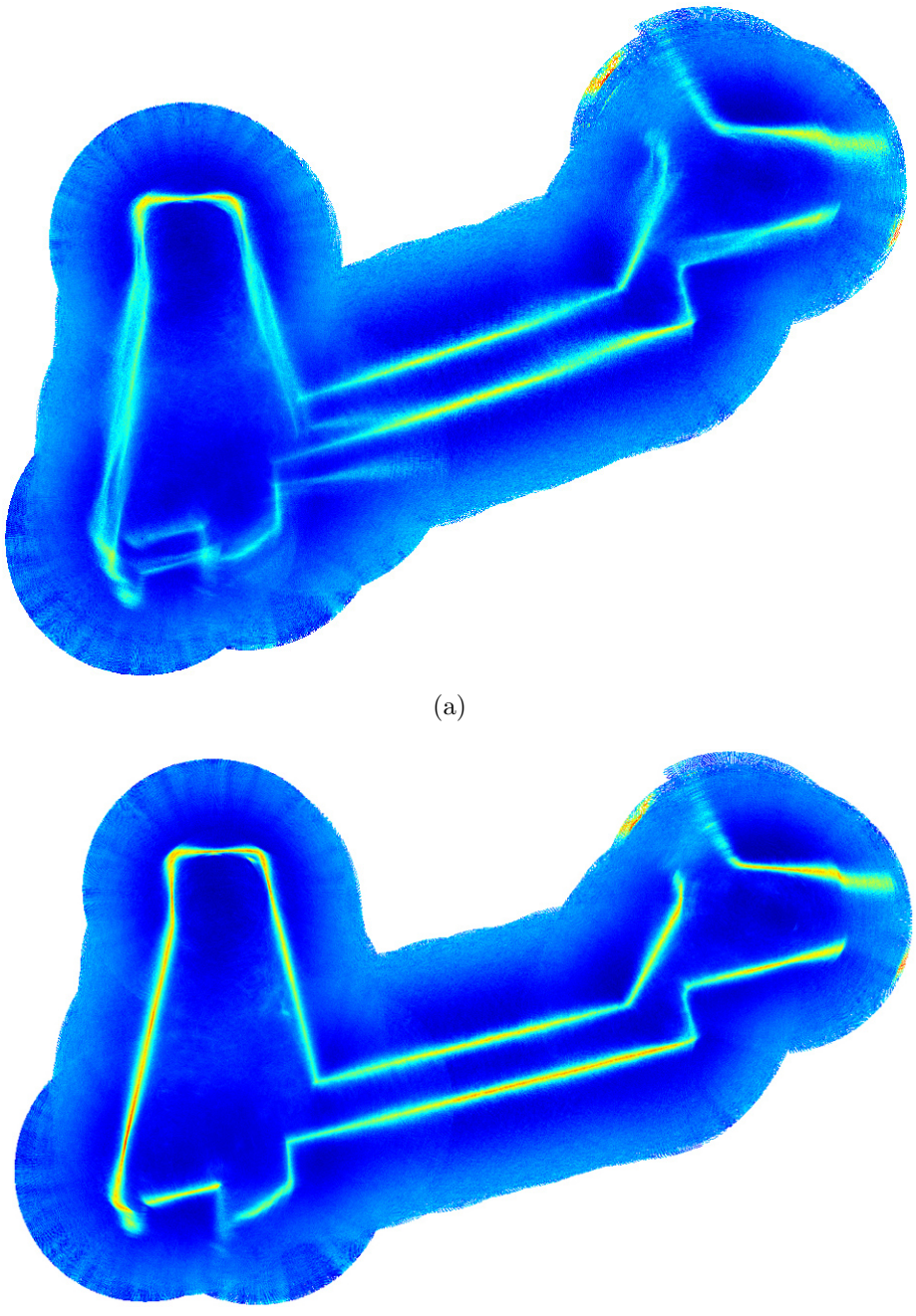


Figure 6.18: Error plots (2σ bounds) for the resulting estimated trajectory after the local map joining. The DGPS data has been used as ground truth.



(a)

(b)

Figure 6.19: Acoustic maps obtained after an averaged composition of the sonar readings along different trajectories. (a) The filter executed with only the input from the dead-reckoning sensors. (b) SLAM estimated trajectory.

Chapter 7

Conclusion

This chapter concludes the work presented throughout this document. It first summarizes the thesis by reviewing the contents described in each chapter. It then points out the research contributions extracted from the proposals and the experiments. In addition, all aspects which have not been accomplished as well as some interesting future research issues are commented on in the future work section. Then the research framework in which this thesis was achieved is described. Finally, the publications related to this work are listed.

7.1 Summary

The navigation problem is one of the most crucial problems that needs to be addressed to obtain truly autonomous vehicles. The SLAM approach represents the ultimate navigation solution since it makes it possible to localize a vehicle and simultaneously map the environment without the need of external devices or previous knowledge of the scenario.

Chapter 2 presents the elemental principles behind the SLAM solution and briefly overviews the history and recent developments done by the research community on the topic. Then, the chapter focuses on the use of SLAM systems in underwater environments, introducing the different sensorial options based on acoustic devices available and finally reviewing the most remarkable work done in the field. This review leads to interesting observations such as the tendency to use vehicles equipped with high cost systems like electronically scanned devices, or the fact that the main part of the works revolve around the possible application of SLAM systems in natural underwater environments. In contrast with this, the work presented throughout this dissertation is centered on the use of mechanically scanned sonars and the exploration of new application domains like those found in

man-made environments. Chapter 3 presents the Ictineu AUV, the research vehicle developed parallel to the realization of this thesis, that was employed for the experimental work. The structure, the sensor suite and the software architecture are described as well as plans for future upgrades. The introductory part of this document ends with Chapter 4, where the principles of operation of MSIS regarding the generation and interpretation of acoustic images are explained. The principal issues related with the use of this type of sensors are described and identified as problems to be addressed by the proposed navigation algorithms.

The second part of this work presents different approaches for navigating in man-made underwater environments endorsed with many experiments carried out with real sensor data. Chapter 5 describes three localization methods of increasing complexity relying on different sensors and an *a priori map* of the environment. The first is based on the use of an adapted version of the Hough transform to vote for the vehicle's position using the measurements from an MSIS and the heading from a compass. The second introduces a Kalman filter and the use of a DVL sensor to estimate the vehicle's velocities. The measurements from the MSIS are individually contrasted with the map to perform position updates. Finally, the third method combines the other two by setting up a Kalman filter which receives updates from an improved version of the Hough-based voting scheme. The experimental demonstration includes a comparison of the methods with a test performed in the CIRS water tank. An additional example is presented to show the performance of the first approach as the navigation system of the Ictineu AUV during the SAUC-E competition. Chapter 6 culminates this work with the proposal of a SLAM system for navigation in structured environments composed of rectilinear walls. This system presents a line feature extraction algorithm capable of dealing with the issues associated with the operation of MSISs. It uses a voting scheme to look for new features and analyzes the imprint left in the acoustic images to determine its uncertainty. The SLAM consists of an EKF-based implementation of the stochastic map. A constant velocity kinematics model predicts the vehicle's motion while the measurements from a DVL, a compass and a pressure sensor update the estimate. The observations from the line feature extraction algorithm are incorporated into the map building process and, as a consequence, the vehicle's position is improved. In addition, the problems associated with dealing with large scenarios have also been addressed and solved through the implementation of a local map building procedure. The final part of the chapter proves the capacity of the proposal with two different tests. The first evaluates the performance of the SLAM approach in comparison with the previously proposed localization methods by means of the CIRS water tank experiment. The sec-

ond experiment is more ambitious and realistic. A long run performed in an abandoned marina serves as a test not only of the SLAM approach but also the local map building method.

7.2 Contributions

This thesis work has accomplished the proposed goal of developing a SLAM approach for an AUV to achieve localization in man-made underwater environments using an MSIS as the principal sensor. In the development of this goal, various research contributions were achieved. These contributions are listed below:

Application domain: Most of the works published on underwater SLAM present systems operating in natural environments and use point features to build the map. To the best of the author's knowledge, the approach presented here is the first work on underwater SLAM focused on the operation in man-made structured environments and in the use of line features for the representation of underwater scenarios. This contribution opens the door to the use of SLAM systems in new application domains.

Localization with an *a priori* map: Three different methods for localization with an *a priori* map have been developed and tested. Although the use of Kalman filters and voting schemes in localization is not new, the particular treatment given to the MSIS data and its application in man-made underwater environments adds value to the approach and represents a contribution to the field.

Feature extraction from MSIS data: An important contribution of this work is the development of a method to extract features from acoustic underwater images acquired with an MSIS. The method deals with the continuous arrival of measurements, undistorts the data affected by the vehicle's motion and determines the uncertainty of the observed features from the imprint left in the acoustic images. This method makes it possible to use MSIS for SLAM as a lower cost alternative to the electronically scanned sonars.

Simultaneous localization and mapping: The number of published SLAM approaches for submersibles operating in underwater environments is still small. For this reason, developing a new algorithm for an unattempted application domain and demonstrating its operation with real sensor data constitutes an important contribution of this thesis.

Dataset acquisition: Producing a dataset is not an easy task when operating with underwater vehicles. The sensors and equipment are expensive and the working conditions are difficult. Moreover, obtaining ground truth data to validate the navigation algorithms is not common. An additional contribution of this thesis is making the abandoned marina dataset publicly available for the research community. The dataset can be downloaded from [Ribas, 2006].

7.3 Future Work

During the development of this research work, new problems and topics of interest for future research have arisen. The following points are the ones which have been found as the most logical lines to continue this research:

Development of the localization systems: Although the proposed localization systems have shown promising results, further work needs to be done to improve the reliability of the system before it is implemented in a real vehicle. The current methods are adequate only for planar environments or those composed only of vertical walls which have a constant section independent of the vehicle's depth. Producing a tridimensional *a priori* map of the scenario would make it possible to define the characteristics of the voting space depending on a particular operational depth estimated through pressure sensor measurements. A second issue to be addressed is to guarantee the safe operation of the vehicle in the absence of reliable sensor measurements particularly while operating very close to walls (strong compass distortions) or near the bottom (DVL is unable to produce velocity measurements). Of course, the integration of better (and more expensive) sensors such as fiber optic gyros can solve most of these problems. However, exploring alternatives based on the introduction of the vehicle's dynamics in the prediction part of the filter would be interesting and inexpensive.

Representation of line features: The line feature extraction algorithm for the proposed SLAM system provides polar observations (ρ and θ parameters) which are referenced to the sensor base frame. The author believes that this representation is the most adequate given the characteristics of the sensor and the measurement estimation process. However, this is not so true when the features are introduced into the map. Their base reference frame is changed from the one in the sensor to the one in which the map is represented. When the reference

frame is placed far from the current position of a particular line segment (large ρ value), even the smaller correction in the angle (θ value) may produce large displacements in the segment position and hence, an awkward representation of the map. Although this was sufficient while operating in scenarios of reduced dimensions, at the present time and having the testing in larger environments in perspective, this has become one of the principal weak points of the proposed SLAM approach. Although the use of a smaller local map has shown potential in mitigating this effect, it would be prefereable to implement a better method to represent the line features. The plans to improve the SLAM algorithm shortly after the presentation of this document include the implementation of the SP map [Castellanos and Tardós, 1999] as a worthwhile candidate to solve this problem.

Extended feature typology: Line features are characteristic of man-made environments. Although the use of lines has shown to be sufficient for the presented test scenarios, introducing new types of features can offer a better and richer representation of the environment. The proposed feature extraction algorithm can be easily adapted to other types of features as long as a suitable parametrization is possible. One of the simplest options is the detection of corners at the intersection of two walls. The detection of curves or planes are other candidates, although their parametrization requirements would entail the use of higher-dimensional voting spaces. This can even lead to obtaining tridimensional map representations of the explored spaces.

New scenarios: Better and improved algorithms should be tested in new and more challenging scenarios. Operating in real scenarios populated with moored boats, piers, breakwaters and other common elements may offer the possibility of producing new types of features but also generate new problems. For instance, the presence of traffic in a harbor-like scenario would require more sophisticated data association techniques to discriminate static elements from moving boats.

7.4 Research Framework

The results and conclusions presented in this thesis have been possible after the realization of countless tests and experiments, which were the fruit of numerous efforts done in the development of the different research vehicles and the necessary software and equipments. All the work done during the evolution of this thesis is summarized here with references to the most relevant

research publications done by the author. The complete list of publications can be consulted in the next section.

At the beginning of this thesis in the year 2003, there were two research platforms in the Underwater Robotics laboratory at the University of Girona. The first was the URIS AUV [AMI'04], a robot of reduced dimensions designed to operate under laboratory conditions in a small tank. The second was the GARBI AUV, a larger vehicle for operation in real environments which at the moment was undergoing an intense remodeling and upgrade process. One of the new systems to be installed in the GARBI was the recently purchased Argonaut DVL sensor. In [WESIC'03], it was presented with preliminary work towards the development of an EKF-based navigation system to integrate the different DVL measurements with the predictions of a dynamic model. In parallel to the long development period of the GARBI, many works were carried out with the URIS vehicle and particularly, with a localization system developed for the small experimental tank [IBPRIA'03]. The system comprised a down-looking camera mounted on the robot and a coded pattern placed on the bottom of the tank. The incorporation of an EKF to improve its position estimates was also studied [MCMC'03]. A similar EKF was later implemented on an image mosaicking approach tested with the same URIS vehicle over a staged underwater scenario [IROS'03]. On mid-2004, many datasets were acquired on the Costa Brava, near Colera (Spain), on the context of a new research line on underwater SLAM. Unfortunately, the upgrading of the GARBI still had not been completed at the time. For this reason, different sensors (including a GPS, the DVL and a new sensor, the MSIS) had to be mounted on a metallic structure and attached to a small boat to perform the experiments. The analysis of the resulting datasets and the development of the first SLAM system relying on natural features was done during a research stay at the University of Zaragoza, under the supervision of Prof. Jose Neira. The result of this work was presented in [MASTER'05]. In 2006, the GARBI was operative and the new installations at the CIRS were completely functional. This same year, a group of students entered the SAUCE 06 competition and, in a short period of time, developed the Ictineu AUV [CCIA'06]. The author acted as team leader and took part in the construction of the new prototype, as well as in the design of different localization approaches for the competition [MCMC'06, ICRA'07], which were later improved with a new proposal [CAMS'07]. Simultaneously, the work on SLAM continued with the development of a system for structured environments which was tested in the CIRS water tank [IROS'06]. The next year, a new approach for feature extraction was developed [IBPRIA'07, IAV'07] and a new dataset was acquired in the abandoned marina scenario using the Ictineu AUV. These results served to

demonstrate an improved SLAM approach [IROS'07, MARTECH'07]. Later, the introduction of a strategy for building sequences of local maps improved the final results [JFR'08].

7.5 Related Publications

Simultaneous Localization and Mapping

[JFR'08] D. Ribas, P. Ridao, J.D. Tardós and J. Neira. Underwater SLAM in man made structured environments. Journal of Field Robotics, *accepted for publication*, 2008.

[MARTECH'07] D. Ribas, P. Ridao, J. Neira and J.D. Tardós. Underwater SLAM for man-made environments. 2nd Congrés Internacional sobre Tecnologia Marina (Martech'07), Vilanova i la Geltrú, Spain, November 2007.

[IROS'07] D. Ribas, P. Ridao, J.D. Tardós and, J. Neira. Underwater SLAM in a marina environment. IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, USA, October 2007.

[IROS'06] D. Ribas, J. Neira, P. Ridao and J.D. Tardós. SLAM using an imaging sonar for partially structured environments. IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, October 2006.

[MASTER'05] D. Ribas. Towards Simultaneous Localization & Mapping for an AUV using an imaging sonar. Master thesis, University of Girona, Spain, 2005.

Feature extraction from sonar images

[IAV'07] D. Ribas, P. Ridao, J. Neira and J.D. Tardós. A method for extracting lines and their uncertainty from acoustic underwater images for SLAM. 6th IFAC Symposium on Intelligent Autonomous Vehicles, Toulouse, France, September 2007.

[IBPRIA'07] D. Ribas, P. Ridao, J. Neira and J.D. Tardós. Line extraction from mechanically scanned imaging sonar. 3rd Iberian Conference on Pattern Recognition and Image Analysis. Lecture Notes in Computer Science, 4477:322–329, 2007.

Localization

[CAMS'07] G. García de Marina, D. Ribas and P. Ridao. A global localization system for structured environments using an imaging sonar. IFAC Conference on Control in Marine Systems, Bol, Croatia, September 2007.

[ICRA'07] D. Ribas, N. Palomeras, P. Ridao, M. Carreras and E. Hernàndez. Ictineu AUV wins the first SAUC-E competition. IEEE International Conference on Robotics and Automation, Roma, Italy, April 2007.

[CCIA'06] E. Hernàndez, P. Ridao, M. Carreras, D. Ribas and N. Palomeras. Ictineu AUV, un Robot per a Competir. 9th Congrés Català d'Intel.ligència Artificial, Perpignan, France, October 2006.

[MCMC'06] D. Ribas, J. Neira, P. Ridao and J.D. Tardós. AUV localization in structured underwater environments using an a priori map. 7th IFAC Conference on Manoeuvring and Control of Marine Crafts, Lisboa, Portugal, September 2006.

[IBPRIA'03] M. Carreras, P. Ridao, J. Batlle, and D. Ribas. High-accuracy localization of an underwater robot in a structured environment using computer vision. 2nd Iberian Conference on Pattern Recognition and Image Analysis. Lecture Notes in Computer Science, 2652:150–157, 2003.

[IROS'03] R. García, T. Nicosevici, P. Ridao, and D. Ribas. Towards a real-time vision-based navigation system for a small-class UUV. IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, USA, October 2003.

[MCMC'03] D. Ribas, P. Ridao, M. Carreras, and X. Cufí. An EKF vision-based navigation of an UUV in a structured environment. 6th IFAC Conference on Manoeuvring and Control of Marine Crafts, Girona, Spain, September 2003.

[WESIC'03] D. Ribas, P. Ridao, X. Cufí, and A. El-fakdi. Towards a DVL-based navigation system for an underwater robot. 4th Workshop on European Scientific and Industrial Collaboration, Miskolc, Hungary, May 2003.

Related work in underwater robotics

[IJC'07] M. Carreras, N. Palomeras, P. Ridao, and D. Ribas. Design of a mission control system for an AUV. International Journal of Control, 80(7):993–1007, July 2007.

[OCEANS'04] P. Ridao, E. Batlle, D. Ribas, and M. Carreras. NEPTUNE: A HIL simulator for multiple UUVs. Oceans 04 MTS/IEEE, Kobe, Japan, November 9-12 2004.

[WAF'04] P. Ridao, D. Ribas, E. Batlle, and E. Hernàndez. Simulation of physical agents. An application to underwater robots. V Workshop on Physical Agents, Girona, Spain, March 2004.

[ISCCSP'04] P. Ridao, M. Carreras, D. Ribas, and A. El-Fakdi. Graphical simulators for AUV development. First International Symposium on Control, Communications and Signal Processing, Hammamet, Tunisia, March 2004.

[AMI'04] J. Batlle, P. Ridao, R. García, M. Carreras, X. Cufí, A. El-Fakdi, D. Ribas, T. Nicosevici, E. Batlle, G. Oliver, A. Ortiz and J. Antich. URIS: Underwater Robotic Intelligent System. Automation for the Maritime Industries, chapter 11, pages 177–203. Instituto de Automática Industrial, Consejo Superior de Investigaciones Científicas, first edition, 2004.

Appendix A

The Kalman filter

The Kalman filter is a recursive data processing algorithm which addresses the general problem of estimating the state of a stochastic system using a model of the system and a set of sensor measurements that are functions of the state. This appendix presents the equations for the linear and non-linear formulations of the Kalman filter. A more detailed description on this topic can be found in [Kalman, 1960; Maybeck, 1982].

A.1 The linear Kalman Filter

A.1.1 Linear system models

The state vector $\mathbf{x}$ to be estimated describes the state of a discrete-time controlled process governed by a linear stochastic difference equation. This equation is generally denominated as the process model:

$$\mathbf{x}(k) = \mathbf{A}\mathbf{x}(k-1) + \mathbf{B}\mathbf{u}(k) + \mathbf{n}(k-1),$$

where $\mathbf{A}$ is a matrix that relates the state at $k-1$ to the actual state at time k, $\mathbf{B}$ is a matrix determining the effect that the control input $\mathbf{u}$ produces on the evolution to the actual state and finally, $\mathbf{n}$ is a noise representing the process uncertainty which is assumed independent, white, and with a Gaussian probability distribution of covariance $\mathbf{Q}$:

$$\begin{aligned} E\left[\mathbf{n}(k)\right] &= \mathbf{0}, \\ E\left[\mathbf{n}(k)\mathbf{n}(j)^T\right] &= \delta_{kj}\mathbf{Q}(k), \end{aligned}$$

At discrete intervals, the sensors provide observations of the system's state. This process is described with the measurement model:

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{m}(k),$$

where $\mathbf{H}$ is a matrix relating measurement $\mathbf{z}$ to state $\mathbf{x}$ and $\mathbf{m}$ is an independent white Gaussian noise with covariance $\mathbf{R}$ that represents the measurement's uncertainty.

$$\begin{aligned} E\left[\mathbf{m}(k)\right] &= \mathbf{0}, \\ E\left[\mathbf{m}(k)\mathbf{m}(j)^T\right] &= \delta_{kj}\mathbf{R}(k). \end{aligned}$$

A.1.2 The Discrete Kalman filter equations

The objective of the filter is to obtain an estimate of the system's state represented by the mean $\hat{\mathbf{x}}$ and the variance $\mathbf{P}$ of the state distribution:

$$\begin{aligned} E\left[\mathbf{x}(k)\right] &= \hat{\mathbf{x}}(k), \\ E\left[(\mathbf{x}(k) - \hat{\mathbf{x}}(k))(\mathbf{x}(k) - \hat{\mathbf{x}}(k))^T\right] &= \mathbf{P}(k). \end{aligned}$$

The recursive estimation process of the Kalman filter is divided into two parts: the prediction and the correction. The prediction step projects the estimates of the state vector and its error covariances ahead in time by means of the stated process model. The equations responsible for this are:

$$\begin{aligned} \hat{\mathbf{x}}(k|k-1) &= \mathbf{A}\mathbf{x}(k-1) + \mathbf{B}\mathbf{u}(k), \\ \mathbf{P}(k|k-1) &= \mathbf{A}\mathbf{P}(k-1)\mathbf{A}^T + \mathbf{Q}, \end{aligned}$$

where $[\hat{\mathbf{x}}(k|k-1), \mathbf{P}(k|k-1)]$ is the estimated prediction of the current state $\mathbf{x}$(k) obtained from the estimate at time k, the control input $\mathbf{u}(k)$ and the model defined by $\mathbf{A}$ and $\mathbf{B}$. The increment of the estimate uncertainty inherent to a prediction process is reflected with the addition of the term $\mathbf{Q}$ that corresponds to the covariance of the noise in the process model. The next step is to update this estimate by adding the information provided by a sensor measurement $\mathbf{z}(k)$. This is achieved with the measurement update equations of the Kalman filter:

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \hat{\mathbf{x}}(k|k-1) + \mathbf{K}\boldsymbol{\nu}, \\ \mathbf{P}(k) &= (\mathbf{I} - \mathbf{K}\,\mathbf{H})\mathbf{P}(k|k-1), \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{\nu} &= \mathbf{z}(k) - \mathbf{H}\hat{\mathbf{x}}(k|k-1), \\ \mathbf{S} &= \mathbf{H}\mathbf{P}(k|k-1)\mathbf{H}^T + \mathbf{R}, \\ \mathbf{K} &= \mathbf{P}(k|k-1)\mathbf{H}^T\mathbf{S}^{-1}. \end{aligned}$$

The term $\boldsymbol{\nu}$ represents the discrepancy between the actual sensor measurement $\mathbf{z}$ and the prediction of this same measurement obtained with the measurement model $\mathbf{H}\hat{\mathbf{x}}$, $\mathbf{S}$ being its corresponding covariance. This is necessary to calculate $\mathbf{K}$, the Kalman gain, which is chosen to correct the estimate and minimize the error covariance $\mathbf{P}$ after the update.

A.2 The Extended Kalman Filter

A.2.1 Non-linear system models

The extended Kalman filter is a version of the Kalman filter that can deal with systems governed by non-linear stochastic difference equations. In this situation, a non-linear process model is defined as:

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k), \mathbf{n}(k-1)),$$

while a non-linear measurement model is represented as:

$$\mathbf{z}(k) = h(\mathbf{x}(k), \mathbf{m}(k)),$$

$\mathbf{n}$ and $\mathbf{m}$ being analogous to the process and measurement noises defined in the linear version of the filter in Section A.1.1.

A.2.2 The Discrete Extended Kalman Filter equations

The extended Kalman filter deals with the non-linearities of the system by performing linearizations for the current mean and covariance. The equations for the two-step recursive estimation process are similar to those from the Kalman filter:

$$\begin{aligned} \hat{\mathbf{x}}(k|k-1) &= f(\hat{\mathbf{x}}(k-1), \mathbf{u}(k), 0), \\ \mathbf{P}(k|k-1) &= \mathbf{F}(k)\mathbf{P}(k-1)\mathbf{F}^T(k) + \mathbf{W}(k)\mathbf{Q}\mathbf{W}^T(k). \end{aligned}$$

The $\mathbf{F}$ and $\mathbf{W}$ Jacobian matrices are responsible for the linearization. They contain the partial derivatives of the f function with respect to the state $\mathbf{x}$ and the process noise $\mathbf{n}$:

$$\begin{aligned} \mathbf{F}(k) &= \frac{\partial f}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k|k-1), \mathbf{u}(k), \mathbf{0}) \\ \mathbf{W}(k) &= \frac{\partial f}{\partial \mathbf{n}}(\hat{\mathbf{x}}(k|k-1), \mathbf{u}(k), \mathbf{0}) \end{aligned}$$

The measurement update equations are also adapted to the use of non-linear measurement equations:

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \hat{\mathbf{x}}(k|k-1) + \mathbf{K}\boldsymbol{\nu}, \\ \mathbf{P}(k) &= (\mathbf{I} - \mathbf{K}\,\mathbf{H}(k))\mathbf{P}(k|k-1), \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{\nu} &= \mathbf{z}(k) - h(\hat{\mathbf{x}}(k|k-1), 0), \\ \mathbf{S} &= \mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}^T(k) + \mathbf{V}(k)\mathbf{R}\mathbf{V}^T(k), \\ \mathbf{K} &= \mathbf{P}(k|k-1)\mathbf{H}^T(k)\mathbf{S}^{-1}. \end{aligned}$$

Again, the Jacobians $\mathbf{H}$ and $\mathbf{V}$ are necessary to linearize the measurement function h:

$$\begin{aligned} \mathbf{H}(k) &= \frac{\partial h}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k|k-1), \mathbf{0}), \\ \mathbf{V}(k) &= \frac{\partial h}{\partial \mathbf{m}}(\hat{\mathbf{x}}(k|k-1), \mathbf{0}). \end{aligned}$$

Appendix B

Transformations in 2D

In [Smith et al., 1990] two operations were presented representing the most frequently encountered spatial relationships in stochastic mapping applications. These are the inversion and compounding transformations, represented by the operators $\ominus$ and $\oplus$:

$$\begin{aligned} \mathbf{x}_C^A &= \mathbf{x}_B^A \oplus \mathbf{x}_C^B, \\ \mathbf{x}_C^A &= \ominus \mathbf{x}_A^C. \end{aligned}$$

Here, these operators will be described together with two additional compounding operators for transforming the references of point and line features.

B.1 Inversion

Given a spatial transformation (location of a reference B relative to reference A):

$$\mathbf{x}_B^A = \begin{bmatrix} x_1 \\ y_1 \\ \phi_1 \end{bmatrix}.$$

The location of A relative to B can be described by the inversion operation $\ominus$:

$$\mathbf{x}_A^B = \ominus \mathbf{x}_B^A = \begin{bmatrix} -x_1 \cos\phi_1 - y_1 \sin\phi_1 \\ x_1 \sin\phi_1 - y_1 \cos\phi_1 \\ -\phi_1 \end{bmatrix}.$$

The Jacobian of the inversion operation is:

$$\mathbf{J}_{\ominus} = \begin{bmatrix} -\cos\phi_1 & -\sin\phi_1 & -x_1 \sin\phi_1 - y_1 \cos\phi_1 \\ \sin\phi_1 & -\cos\phi_1 & x_1 \cos\phi_1 + y_1 \sin\phi_1 \\ 0 & 0 & -1 \end{bmatrix}.$$

Therefore, given the estimated mean and covariance of the spatial transformation:

$$\begin{aligned} E\left[\mathbf{x}_{B}^{A}\right] &= \hat{\mathbf{x}}_{B}^{A}, \\ E\left[(\mathbf{x}_{B}^{A}-\hat{\mathbf{x}}_{B}^{A})(\mathbf{x}_{B}^{A}-\hat{\mathbf{x}}_{B}^{A})^{T}\right] &= \mathbf{P}_{B}^{A}. \end{aligned}$$

The estimated location of A relative to B can be described as the inversion:

$$\hat{\mathbf{x}}_{A}^{B} = \ominus\hat{\mathbf{x}}_{B}^{A},$$

With associated covariance calculated as:

$$\mathbf{P}_{A}^{B} \simeq \mathbf{J}_{\ominus}\mathbf{P}_{B}^{A}\mathbf{J}_{\ominus}^{T}.$$

B.2 Compounding

Given two spatial transformations (reference B relative to reference A and reference C relative to reference B):

$$\mathbf{x}_{B}^{A} = \begin{bmatrix} x_1 \\ y_1 \\ \phi_1 \end{bmatrix}, \qquad \mathbf{x}_{C}^{B} = \begin{bmatrix} x_2 \\ y_2 \\ \phi_2 \end{bmatrix}.$$

The location of C relative to A can be described by the compounding operation as:

$$\mathbf{x}_{C}^{A} = \mathbf{x}_{B}^{A} \oplus \mathbf{x}_{C}^{B} = \begin{bmatrix} x_1 + x_2\cos\phi_1 - y_2\sin\phi_1 \\ y_1 + x_2\sin\phi_1 + y_2\cos\phi_1 \\ \phi_1 + \phi_2 \end{bmatrix}.$$

Two Jacobian matrices are necessary to linearize the compounding operation with respect to each one of the two spatial transformations $\mathbf{x}_{B}^{A}$ and $\mathbf{x}_{C}^{B}$:

$$\begin{aligned} \mathbf{J}_{1\oplus} &= \begin{bmatrix} 1 & 0 & -x_2\sin\phi_1 - y_2\cos\phi_1 \\ 0 & 1 & x_2\cos\phi_1 - y_2\sin\phi_1 \\ 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{J}_{2\oplus} &= \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 & 0 \\ \sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned}$$

So, given the estimated mean and covariance of the spatial transformations $(\hat{\mathbf{x}}_{B}^{A}, \mathbf{P}_{B}^{A})$ and $(\hat{\mathbf{x}}_{C}^{B}, \mathbf{P}_{C}^{B})$, the estimated location of C relative to A can be described as the compounding:

$$\hat{\mathbf{x}}_{C}^{A} = \hat{\mathbf{x}}_{B}^{A} \oplus \hat{\mathbf{x}}_{C}^{B}.$$

with associated covariance approximated as:

$$\mathbf{P}_{C}^{A} \simeq \mathbf{J}_{1\oplus}\mathbf{P}_{B}^{A}\mathbf{J}_{1\oplus}^{T} + \mathbf{J}_{2\oplus}\mathbf{P}_{C}^{B}\mathbf{J}_{2\oplus}^{T}.$$

B.3 Compounding point features

Given the location of point feature P relative to reference B:

$$\mathbf{x}_{P}^{B} = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}.$$

In a similar manner as mentioned before, the location of P relative to reference A can be described by the compounding operation for a point:

$$\mathbf{x}_{P}^{A} = \mathbf{x}_{B}^{A} \oplus \mathbf{x}_{P}^{B} = \begin{bmatrix} x_1 + x_2\cos\phi_1 - y_2\sin\phi_1 \\ y_1 + x_2\sin\phi_1 + y_2\cos\phi_1 \end{bmatrix}.$$

The Jacobians of this transformation are:

$$\begin{aligned} \mathbf{J}_{1\oplus} &= \begin{bmatrix} 1 & 0 & -x_2\sin\phi_1 - y_2\cos\phi_1 \\ 0 & 1 & x_2\cos\phi_1 - y_2\sin\phi_1 \end{bmatrix}, \\ \mathbf{J}_{2\oplus} &= \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 \\ \sin\phi_1 & \cos\phi_1 \end{bmatrix}. \end{aligned}$$

Again, given the estimated mean and covariance of the spatial transformation $(\hat{\mathbf{x}}_{B}^{A}, \mathbf{P}_{B}^{A})$ and the point $(\hat{\mathbf{x}}_{P}^{B}, \mathbf{P}_{P}^{B})$, the estimated location of P relative to A can be described as the composition transformation:

$$\hat{\mathbf{x}}_{P}^{A} = \hat{\mathbf{x}}_{B}^{A} \oplus \hat{\mathbf{x}}_{P}^{B}$$

and its associated covariance as:

$$\mathbf{P}_{P}^{A} \simeq \mathbf{J}_{1\oplus}\mathbf{P}_{B}^{A}\mathbf{J}_{1\oplus}^{T} + \mathbf{J}_{2\oplus}\mathbf{P}_{P}^{B}\mathbf{J}_{2\oplus}^{T}$$

B.4 Compounding line features

Given the location of line feature L represented in polar coordinates with respect to reference B:

$$\mathbf{x}_{L}^{B} = \begin{bmatrix} \rho_2 \\ \theta_2 \end{bmatrix}.$$

In a similar manner, as mentioned before, the location of L in polar coordinates relative to reference A can be described by the compounding operation for a line:

$$\mathbf{x}_{L}^{A} = \mathbf{x}_{B}^{A} \oplus \mathbf{x}_{L}^{B} = \begin{bmatrix} x_1 \cos(\phi_1 + \theta_2) + y_1 \sin(\phi_1 + \theta_2) + \rho_2 \\ \phi_1 + \theta_2 \end{bmatrix}.$$

The Jacobians of this transformation are:

$$\mathbf{J}_{1\oplus} = \begin{bmatrix} \cos(\phi_1 + \theta_2) & \sin(\phi_1 + \theta_2) & -x_1 \sin(\phi_1 + \theta_2) + y_1 \cos(\phi_1 + \theta_2) \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{J}_{2\oplus} = \begin{bmatrix} 1 & -x_1 \sin(\phi_1 + \theta_2) + y_1 \cos(\phi_1 + \theta_2) \\ 0 & 1 \end{bmatrix}.$$

Again, given the estimated mean and covariance of the spatial transformation $(\hat{\mathbf{x}}_{B}^{A}, \mathbf{P}_{B}^{A})$ and the line $(\hat{\mathbf{x}}_{L}^{B}, \mathbf{P}_{L}^{B})$, the estimated polar parameters of L relative to A can be described as the composition transformation:

$$\hat{\mathbf{x}}_{L}^{A} = \hat{\mathbf{x}}_{B}^{A} \oplus \hat{\mathbf{x}}_{L}^{B},$$

and its associated covariance as:

$$\mathbf{P}_{L}^{A} \simeq \mathbf{J}_{1\oplus} \mathbf{P}_{B}^{A} \mathbf{J}_{1\oplus}^{T} + \mathbf{J}_{2\oplus} \mathbf{P}_{L}^{B} \mathbf{J}_{2\oplus}^{T}.$$

Bibliography

Amat, J., Batlle, J., Montferrer, A., Salvi, J. and Ridao, P.: 1998, Capabilities of GARBI - A low cost underwater vehicle, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Arkin, R. C.: 1998, *Behavior-Based Robotics*, The MIT Press, Cambridge, MA, USA.

Bailey, T. and Durrant-Whyte, H. F.: 2006, Simultaneous localization and mapping (SLAM): Part II, state of the art, *IEEE Robotics and Automation Magazine* **13**(3), 108–117.

Bailey, T., Nebot, E. M., Rosenblatt, J. K. and Durrant-Whyte, H. F.: 2000, Data association for mobile robot navigation: A graph theoretic approach, *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, pp. 2512–2517.

Ballard, D. H.: 1987, *Generalizing the Hough transform to detect arbitrary shapes*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Batlle, J., Nico, T., García, R. and Carreras, M.: 2003, ROV-aided dam inspection: Practical results, *Proceedings of the 6th IFAC Conference on Manoeuvring and Control of Marine Crafts*, Girona, Spain.

Batlle, J., Ridao, P., García, R., Carreras, M., Cufí, X., El-Fakdi, A., Ribas, D., Nicosevici, T. and Batlle, E.: 2004, *URIS: Underwater Robotic Intelligent System*, 1st edn, Instituto de Automática Industrial, Consejo Superior de Investigaciones Científicas, chapter 11, pp. 177–203.

Bosse, M., Newman, P., Leonard, J. J. and Teller, S.: 2004, SLAM in large-scale cyclic environments using the atlas framework, *International Journal of Robotics Research* **23**(12), 1113–1139.

Bosse, M., Newman, P., Leonard, J., Soika, M., Feiten, W. and Teller, S.: 2003, An atlas framework for scalable mapping, *Proceedings of the IEEE*

International Conference on Robotics and Automation, Taipei, Taiwan, pp. 1899–1906.

Burgard, W., Fox, D., Hennig, D. and Schmidt, T.: 1996, Estimating the absolute position of a mobile robot using position probability grids, *Proceedings of the 14th National Conference on Artificial Intelligence*.

Caccia, M., Bruzzone, G. and Veruggio, G.: 2001, Sonar-based guidance of unmanned underwater vehicles, *Advanced Robotics* **15**(5), 551–573.

Carpenter, R. N.: 1998, Concurrent mapping and localization with FLS, *Workshop on Autonomous Underwater Vehicles*, Cambridge, MA, USA, pp. 133–148.

Carreras, M., Batlle, J. and Ridao, P.: 2001, Hybrid coordination of reinforcement learning-based behaviors for AUV control, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, Hawaii, USA.

Carreras, M., Palomeras, N., Ridao, P. and Ribas, D.: 2007, Design of a mission control system for an AUV, *International Journal of Control* **80**(7), 993–1007.

Carreras, M., Ridao, P., Garcia, R. and Nicosevici, T.: 2003, Vision-based localization of an underwater robot in a structured environment, *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei, Taiwan.

Carreras, M., Yuh, J., Batlle, J. and Ridao, P.: 2005, A behavior-based scheme using reinforcement learning for autonomous underwater vehicles, *IEEE Journal of Oceanic Engineering* **30**(2), 416–427.

Castellanos, J. A., Neira, J. and Tardós, J. D.: 2004, Limits to the consistency of EKF-based SLAM, *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal.

Castellanos, J. A. and Tardós, J. D.: 1999, *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*, Kluwer Academic Publishers, Boston, MA, USA.

Castellanos, J., Montiel, J., Neira, J. and Tardós, J.: 1999, The SPmap: a probabilistic framework for simultaneous localization and map building, *IEEE Transactions on Robotics and Automation* **15**(5), 948–953.

Chatila, R. and Laumond, J. P.: 1985, Position referencing and consistent world modeling for mobile robots, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 138–143.

Choset, H.: 1996, *Sensor Based Motion Planning: The Hierarchical Generalized Voronoi Graph*, PhD thesis, California Institute of Technology.

Clemente, L., Davison, A. J., Reid, I. D., Neira, J. and Tardós, J. D.: 2007, Mapping large loops with a single hand-held camera, *Proceedings of Robotics Science and Systems*, Atlanta, GA, USA.

Cox, I. J.: 1991, Blanche - an experiment in guidance and navigation of an autonomous robot vehicle, *IEEE Transactions on Robotics and Automation* **7**(2), 193–204.

Csorba, M.: 1997, *Simultaneous Localisation and Map Building*, PhD thesis, University of Oxford.

Daniel, S., Leannec, F. L., Roux, C., Soliman, B. and Maillard, E. P.: 1998, Side-scan sonar image matching, *IEEE Journal of Oceanic Engineering* **23**(3), 245–259.

Davison, A. J., Cid, Y. G. and Kita, N.: 2004, Real-time 3D SLAM with wide-angle vision, *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal.

Dissanayake, G., Newman, P., Clark, S., Durrant-Whyte, H. F. and Csorba, M.: 2001, A solution to the simultaneous localization and map building (SLAM) problem, *IEEE Transactions on Robotics and Automation* **17**(3), 229–241.

DSTL: 2006, Student Autonomous Underwater Challenge - Europe (SAUC-E), `http://www.dstl.gov.uk/news_events/competitions/sauce/index.php`. [Online; accessed 12-July-2007].

Duda, R. O. and Hart, P. E.: 1972, Use of the Hough transformation to detect lines and curves in pictures, *Communications of the ACM* **15**(1), 11–15.

Durrant-Whyte, H. F.: 1988, Uncertain geometry in robotics, *IEEE Journal of Robotics and Automation* **4**(1), 23–31.

Durrant-Whyte, H. F. and Bailey, T.: 2006, Simultaneous localization and mapping (SLAM): Part I, the essential algorithms, *IEEE Robotics and Automation Magazine* **13**(2), 99–108.

Durrant-Whyte, H. F., Rye, D. and Nebot, E.: 1995, Localisation of automatic guided vehicles, *Proceedings of the 7th International Symposium on Robotics Research*, pp. 613–625.

Elfes, A.: 1987, Sonar-based real-world mapping and navigation, *IEEE Journal of Robotics and Automation* **3**(3), 249–265.

Estrada, C., Neira, J. and Tardós, J. D.: 2005, Hierarchical SLAM: real-time accurate mapping of large environments, *IEEE Transactions on Robotics* **21**(4), 588–596.

Eustice, R., Pizarro, O. and Singh, H.: 2004, Visually augmented navigation in an unstructured environment using a delayed state history, *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans, USA.

Eustice, R., Singh, H., Leonard, J., Walter, M. and Ballard, R.: 2005, Visually navigating the RMS titanic with SLAM information filters, *Proceedings of Robotics Science and Systems*, Cambridge, MA, USA.

Fairfield, N., Jonak, D., Kantor, G. A. and Wettergreen, D.: 2007, Field results of the control, navigation, and mapping systems of a hovering AUV, *Proceedings of the 15th International Symposium on Unmanned Untethered Submersible Technology*, Durham, NH, USA.

Fairfield, N., Kantor, G. and Wettergreen, D.: 2007b, Real-time SLAM with octree evidence grids for exploration in underwater tunnels, *Journal of Field Robotics* **24**, 3–21.

Ferrer, J., Elibol, A., Delaunoy, O., Gracias, N. and García, R.: 2007, Large-area photo-mosaics using global alignment and navigation data, *Proceedings of the Oceans MTS/IEEE*, Vancouber, Canada.

Fischler, M. A. and Bolles, R. C.: 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.

Folkesson, J., Jensfelt, P. and Christensen, H.: 2005, Vision SLAM in the measurement subspace, *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 30–35.

Fossen, T. I.: 1994, *Guidance and Control of Ocean Vehicles*, John Wiley & Sons Ltd.

Fox, D., Burgard, W., Dellaert, F. and Thrun, S.: 1999, Monte carlo localization: Efficient position estimation for mobile robots, *Proceedings of the 16th National Conference on Artificial Intelligence*.

Fox, D., Burgard, W. and Thrun, S.: 1998, Active markov localization for mobile robots, *Robotics and Autonomous Systems* **25**(3-4), 195–207.

García, R., Nicosevici, T., Ridao, P. and Ribas, D.: 2003, Towards a real-time vision-based navigation system for a small-class UUV, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 1, Las Vegas, USA, pp. 818–823.

García, R., Puig, J., Ridao, P. and Cufí, X.: 2002, Augmented state Kalman filtering for AUV navigation, *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, USA, pp. 4010–4015.

Google Maps: 2008, Abandoned marina, St. Pere Pescador (Spain), `http://maps.google.es/maps?hl=es&ie=UTF8&ll=42.202645,3.10662&spn=0.005182,0.008862&t=h&z=17`. [Online; accessed 3-May-2008].

Gracias, N. R., van der Zwaan, S., Bernardino, A. and Santos-Victor, J.: 2003, Mosaic-based navigation for autonomous underwater vehicles, *IEEE Journal of Oceanic Engineering* **28**(4), 609–624.

Griffiths, G., McPhail, S., Rogers, R. and Meldrum, D.: 1998, Leaving and returning to harbour with an autonomous underwater vehicle, *Proceedings of the Oceanology International*, Brighton, UK.

Grimson, W. E. L.: 1990, *Object Recognition by Computer: The Role of Geometric Constraints*, The MIT Press, Cambridge, MA, USA.

Grisetti, G., Tipaldi, G. D., Stachniss, C., Burgard, W. and Nardi, D.: 2007, Fast and accurate slam with rao-blackwellized particle filters, *Robotics and Autonomous Systems* **55**(1), 30–38.

Guivant, J. E. and Nebot, E. M.: 2001, Optimization of the simultaneous localization and map-building algorithm for real-time implementation, *IEEE Transactions on Robotics and Automation* **17**(3), 242–257.

Gutmann, J. S. and Konolige, K.: 1999, Incremental mapping of large cyclic environments, *Proceedings of the International Symposium on Computational Intelligence in Robotics and Automation*, pp. 318–325.

Heckman, D. B. and Abbot, R. C.: 1973, An acoustic navigation technique, *Proceedings of the Oceans MTS/IEEE*, pp. 591–595.

Huang, S. and Dissanayake, G.: 2007, Convergence and consistency analysis for extended Kalman filter based SLAM, *IEEE Transactions on Robotics* **23**(5), 1036–1049.

Hunt, M., Marquet, W., Moller, D., Peal, K., Smith, W. and Spindel, R.: 1974, An acoustic navigation system, *Technical Report WHOI-74-6*, Woods Hole Oceanographic Institution.

Ictineu: 2007, Ictineu AUV performance during the SAUC-E 06 final run. Video attachment of Ribas et al. [2007], `http://ieeexplore.ieee.org/iel5/4209048/4209049/1248.MM.zip`. [Online; accessed 8-August-2007].

Illingworth, J. and Kittler, J.: 1988, A survey of the Hough transform, *Computer Vision, Graphics, and Image Processing* **44**(1), 87–116.

Imagenex technology corp.: 2002, Sonar theory and applications, `http://www.imagenex.com/sonar_theory.pdf`. [Online; accessed 12-September-2007].

Iovenittl, L., Venturi, M., Albano, G. and Touisi, E.: 1994, Submarine pipeline inspection: the 12 years experience of transmed and future developments, *Proceedings of the 13th International Conference on Offshore Mechanics and Arctic Engineering*, Houston, TX, USA, pp. 149–161.

Kalman, R. E.: 1960, A new approach to linear filtering and prediction problems, *Transactions of the ASME, Journal of Basic Engineering* **82**(Series D), 35–45.

Kim, J. H. and Sukkarieh, S.: 2003, Airborne simultaneous localisation and map building, *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, pp. 406–411.

Kinsey, J., Eustice, R. and Whitcomb, L.: 2006, A survey of underwater vehicle navigation: Recent advances and new challenges, *Proceedings of the 7th IFAC Conference on Manoeuvring and Control of Marine Crafts*, Lisbon, Portugal.

Knight, J., Davison, A. J. and Reid, I.: 2001, Towards constant time SLAM using postponement, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, Hawaii, USA, pp. 406–412.

Kondo, H., Maki, T., Ura, T. and Sakamaki, T.: 2006, AUV navigation based on multi-sensor fusion for breakwater observation, *Proceedings of the 23rd International Symposium on Autonomous Robotics in Construction*, Tokyo, Japan, pp. 72–77.

Kuipers, B. and Byun, Y. T.: 1991, A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations, *Journal of Robotics and Autonomous Systems* **8**, 47–63.

Kuritsky, M. M. and Goldstein, M. S.: 1990, *Autonomous robot vehicles*, Springer-Verlag New York, Inc., New York, NY, USA, chapter Inertial navigation, pp. 96–116.

Leonard, J. J., Carpenter, R. N. and Feder, H. J. S.: 2001, Stochastic mapping using forward look sonar, *Robotica* **19**(5), 467–480.

Leonard, J. J. and Durrant-Whyte, H. F.: 1991, Mobile robot localization by tracking geometric beacons, *IEEE Transactions on Robotics and Automation* **7**(3), 376–382.

Leonard, J. J. and Durrant-Whyte, H. F.: 1992, *Directed Sonar Sensing for Mobile Robot Navigation*, Kluwer Academic Publishers, Norwell, MA, USA.

Leonard, J. J. and Feder, H.: 2001, Decoupled stochastic mapping, *IEEE Journal of Oceanic Engineering* **26**(4), 561–571.

Leonard, J. J. and Feder, H. J. S.: 2000, A computationally efficient method for large-scale concurrent mapping and localization, *in* D. Koditschek and J. Hollerbach (eds), *Proceedings of the 9th International Symposium on Robotics Research*, Springer Verlag, Snowbird, Utah, pp. 169–176.

Leonard, J. J. and Newman, P. M.: 2003, Consistent, convergent and constant-time SLAM, *Proceedings of the International Joint Conference on Artificial Intelligence*, Acapulco, Mexico.

Leonard, J. J. and Rikoski, R. J.: 2000, Incorporation of delayed decision making into stochastic mapping, *Proceedings of the International Symposium on Experimental Robotics*, Vol. 271, pp. 533–542.

Leonard, J. J., Rikoski, R. J., Newman, P. M. and Bosse, M. C.: 2002, Mapping partially observable features from multiple uncertain vantage points, *International Journal of Robotics Research* **21**(10-11), 943–975.

Lim, J. H. and Leonard, J. J.: 2000, Mobile robot relocation from echolocation constraints, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(9), 1035–1041.

Lu, F. and Milios, E.: 1997, Globally consistent range scan alignment for environment mapping, *Autonomous Robots* **4**, 333–349.

Lucas, B. D. and Kanade, T.: 1981, An iterative image registration technique with an application to stereo vision, *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pp. 674–679.

Lucido, L., Opderbecke, J., Rigaud, V., Deriche, R. and Zhang, Z.: 1996, A terrain referenced underwater positioning using sonar bathymetric profiles and multiscale analysis, *Proceedings of the Oceans MTS/IEEE*, Fort Lauderdale, FL, USA.

Mahalanobis, P. C.: 1936, On the generalized distance in statistics, *Proceedings of the National Institute of Science of India*, number 12, pp. 49–55.

Maki, T., Kondo, H., Ura, T. and Sakamaki, T.: 2006, Navigation of an autonomous underwater vehicle for photo mosaicing of shallow vent areas, *Proceedings of the Oceans Asia Pacific*, Singapore.

Maki, T., Kondo, H., Ura, T. and Sakamaki, T.: 2006b, Photo mosaicing of Tagiri shallow vent area by the AUV Tri-Dog 1 using SLAM based navigation scheme, *Proceedings of the Oceans MTS/IEEE*, Boston, MA, USA.

Martins, A., Matos, A., Cruz, N. and Pereira, F. L.: 1999, IES an open system for underwater inspection, *Proceedings of the Oceans MTS/IEEE*, Vol. 2, Seattle, USA, pp. 549–554.

May, M. B.: 1978, Gravity navigation, *Record of the 1978 Position Location and Navigation Symposium*, San Diego, CA, USA, pp. 212–218.

Maybeck, P.: 1982, *Stochastic models, estimation and control*, Vol. 1, Academic Press.

Milne, P. H.: 1983, *Underwater Acoustic Positioning System*, Gulf Publishing Company, Houston, TX, USA.

Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B.: 2002, FastSLAM: A factored solution to the simultaneous localization and mapping problem, *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada.

Moravec, H.: 1988, Sensor fusion in certainty grids for mobile robots, *AI Magazine* **9**(2), 61–74.

Negahdaripour, S. and Xun, X.: 2002, Mosaic-based positioning and improved motion-estimation methods for automatic navigation of submersible vehicles, *IEEE Journal of Oceanic Engineering* **27**(1), 79–99.

Neira, J. and Tardós, J. D.: 2001, Data association in stochastic mapping using the joint compatibility test, *IEEE Transactions on Robotics and Automation* **17**(6), 890–897.

Neira, J., Tardós, J. D. and Castellanos, J. A.: 2003, Linear time vehicle relocation in SLAM, *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, pp. 427–433.

Newman, P., Leonard, J., Tardós, J. D. and Neira, J.: 2002, Explore and return: Experimental validation of real-time concurrent mapping and localization, *Proceedings of the IEEE International Conference on Robotics and Automation*, IEEE, pp. 1802–1809.

Newman, P. M.: 1999, *On the Structure and Solution of the Simultaneous Localisation and Map Building Problem*, PhD thesis, Australian Centre for Field Robotics. The University of Sydney.

Newman, P. M. and Leonard, J. J.: 2003, Pure range-only sub-sea SLAM, *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2, Taipei, Taiwan, pp. 1921–1926.

Newman, P. M., Leonard, J. J. and Rikoski, R. J.: 2003, Towards constant-time SLAM on an autonomous underwater vehicle using synthetic aperture sonar, *Proceedings of the 11th International Symposium on Robotics Research*, Sienna, Italy.

Ni, K., Steedly, D. and Dellaert, F.: 2007, Tectonic SAM: Exact, out-of-core, submap-based SLAM, *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy.

Nie, J., Yuh, J., Kardash, E. and Fossen, T.: 1998, On-board sensor-based adaptative control of small UUVs in very shallow water, *Control Applications in Marina Systems*.

Olson, E., Leonard, J. J. and Teller, S.: 2004, Robust range-only beacon localization, *Autonomous Underwater Vehicles*, pp. 66–75.

Paglia, J. G. and Wyman, W. F.: 1996, DARPA's autonomous minehunting and mapping technologies (AMMT) program: An overview, *Proceedings of the Oceans MTS/IEEE*, Vol. 2, Fort Lauderdale, FL, USA, pp. 794–799.

Palomeras, N., Carreras, M., Ridao, P. and Hernàndez, E.: 2006, Mission control system for dam inspection with an AUV, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, pp. 2551–2556.

Pavlidis, T.: 1982, *Algorithms for graphics and image processing*, Computer Science Press.

Paz, L. M., Piniés, P., Neira, J. and Tardós, J. D.: 2005, Global localization in SLAM in bilinear time, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, pp. 2820–2826.

Petillot, Y., Reed, S. and Bell, J.: 2002, Real time AUV pipeline detection and tracking using side scan sonar and multi-beam echo-sounder, *Proceedings of the Oceans MTS/IEEE*, Vol. 1, Biloxi, MS, USA, pp. 217–222.

Piniés, P. and Tardós, J.: 2007, Scalable SLAM building conditionally independent local maps, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, pp. 3466–3471.

Reed, S., Bell, J. and Petillot, Y.: 2001, Unsupervised segmentation of object shadow and highlight using statistical snakes, *Proceedings of the Generic Oceanographic Array Technology Systems (GOATS) Conference*, La Spezia, Italy.

Reed, S., Petillot, Y. and Bell, J.: 2004, Automated approach to classification of mine-like objects in sidescan sonar using highlight and shadow information, *Proceedings of the IEE Radar, Sonar and Navigation*, Vol. 151, pp. 48–56.

Ribas, D.: 2006, Dataset obtained in an abandoned marina, St. Pere Pescador (Spain), `http://eia.udg.es/%7Edribas`. [Online; accessed 12-July-2007].

Ribas, D., Neira, J., Ridao, P. and Tardós, J. D.: 2006, SLAM using an imaging sonar for partially structured environments, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, pp. 5040–5045.

Ribas, D., Palomer, N., Ridao, P., Carreras, M. and Hernàndez, E.: 2007, Ictineu AUV wins the first SAUC-E competition, *Proceedings of the IEEE International Conference on Robotics and Automation*, Roma, Italy, pp. 151–156.

Ridao, P., Batlle, E., Ribas, D. and Carreras, M.: 2004b, NEPTUNE: A HIL simulator for multiple UUVs, *Proceedings of the Oceans MTS/IEEE*, Vol. 1, Kobe, Japan, pp. 524–531.

Ridao, P., Tiano, A., El-Fakdi, A., Carreras, M. and Zirilli, A.: 2004, On the identification of non-linear models of unmanned underwater vehicles, *Control Engineering Practice* **12**(12), 1483–1499.

Rikoski, R. J. and Leonard, J. J.: 2003, Trajectory sonar perception, *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 1, Taipei, Taiwan, pp. 963–970.

Roman, C. and Singh, H.: 2005, Improved vehicle based multibeam bathymetry using sub-maps and SLAM, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, pp. 3662–3669.

Smith, R. C. and Cheeseman, P.: 1986, On the representation and estimation of spatial uncertainty, *International Journal of Robotics Research* **5**(4), 56–68.

Smith, R., Self, M. and Cheeseman, P.: 1990, Estimating uncertain spatial relationships in robotics, *Autonomous robot vehicles*, Springer-Verlag New York, Inc., New York, NY, USA, pp. 167–193.

Tardós, J. D., Neira, J., Newman, P. and Leonard, J.: 2002, Robust mapping and localization in indoor environments using sonar data, *International Journal of Robotics Research* **21**(4), 311–330.

Tena, I.: 2001, *Enhanced Concurrent Mapping and Localisation Using Forward-looking Sonar*, PhD thesis, Heriot-Watt University.

Tena, I., de Raucourt, S., Petillot, Y. and Lane, D. M.: 2004, Concurrent mapping and localization using sidescan sonar, *IEEE Journal of Oceanic Engineering* **29**(2), 442–456.

Tena, I., Lane, D. and Chantler, M. J.: 1999, A comparison of inter-frame feature measures for robust object classification in sector scan sonar image sequences, *IEEE Journal of Oceanic Engineering* **24**(4), 458–469.

Tena, I., Petillot, Y. and Lane, D. M.: 2003a, Improved AUV navigation using side-scan sonar, *Proceedings of the Oceans MTS/IEEE*, San Diego, CA, USA.

Tena, I., Petillot, Y., Lane, D. M. and Salson, C.: 2001, Feature extraction and data association for AUV concurrent mapping and localisation, *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, pp. 2785–2790.

Tena, I., Reed, S., Petillot, Y., Bell, J. and Lane, D. M.: 2003b, Concurrent mapping and localisation using side-scan sonar for autonomous navigation, *Proceedings of the 13th International Symposium on Unmanned Untethered Submersible Technology*, Durham, NH, USA.

Thrun, S.: 2001, A probabilistic online mapping algorithm for teams of mobile robots, *International Journal of Robotics Research* **20**(5), 335–363.

Thrun, S., Burgard, W. and Fox, D.: 2005, *Probabilistic Robotics*, The MIT Press.

Thrun, S., Liu, Y., Koller, D., Ng, A. Y., Ghahramani, Z. and Durrant-Whyte, H. F.: 2004, Simultaneous localization and mapping with sparse extended information filters, *International Journal of Robotics Research* **23**(7-8), 693–716.

Tyren, C.: 1982, Magnetic anomalies as a reference for ground-speed and map-matching navigation, *Journal of Navigation* **35**(2), 242–254.

Urick, R. J.: 1983, *Principles of underwater sound*, 3rd edition edn, Peninsula Publishing, Los Altos, California.

VICOROB: 2006, Team VICOROB homepage, `http://eia.udg.es/sauce`. [Online; accessed 12-July-2007].

Williams, S. B., Dissanayake, G. and Durrant-Whyte, H. F.: 2002, An efficient approach to the simultaneous localisation and mapping problem, *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, USA, pp. 406–411.

Williams, S. B., Newman, P. M., Rosenblatt, J., Dissanayake, G. and Durrant-Whyte, H.: 2001, Autonomous underwater navigation and control, *Robotica* **19**(5), 481–496.

Williams, S. and Mahon, I.: 2004, Simultaneous localisation and mapping on the great barrier reef, *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2, New Orleans, USA, pp. 1771–1776.