Report on the Research Project

Towards Simultaneous Localization & Mapping for an AUV using an Imaging Sonar

Work presented by:

David Ribas

Supervisors:

Dr. Pere Ridao (University of Girona)
Dr. José Neira (University of Zaragoza)

Universitat de Girona

Department of Electronics, Informatics and Automation

Girona, June 2005

Contents

List of Acronyms

AUV.- Autonomous Underwater Vehicle.

BB.- Branch and Bound.

CML.- Concurrent Mapping and Localization.

CPE.- Constant Pose Estimation.

DVL.- Doppler Velocity Logs.

EIF.- Extended Information Filter.

EKF.- Extended Kalman Filter.

GPS.- Global Positioning System.

IC.- Individual Compatibility.

ICNN.- Individual Compatibility Nearest Neighbor.

IF.- Information Filter.

INS.- Inertial Navigation System.

JCBB.- Joint Compatibility Branch and Bound.

JC.- Joint Compatibility.

KF.- Kalman Filter.

LBL.- Long Base Line.

ROV.- Remotely Operated Vehicle.

SLAM.- Simultaneous Localization And Mapping.

SOG.- Sum of Gaussian.

USBL.- Ultra Short Baseline.

WAAS.- Wide Area Augmentation System.

Chapter 1

Introduction

1.1 Preface

This research project presents the joint work carried out between October 2003 and April 2005 in the Computer Vision and Robotics group at the University of Girona and in the Robotics and Real Time Group at the University of Zaragoza.

This report addresses the navigation problem for an Autonomous Underwater Robot (AUV). Concretely, this work focuses on the Simultaneous Localization And Mapping (SLAM) technique, which has become a very challenging topic in the robotic community over the last few years.

Traditionally, localization and map building problems were treated as independent problems. With an *a priori* known map, a vehicle can recognize the environment through measurements from sensors and determine its location. On the other hand, if the vehicle's position is known, i.e. from external sources such as Global Positioning System (GPS) or Long Base Line (LBL), the data from sensors can be used to generate a map. SLAM techniques try to merge these concepts to obtain a method capable of locating a vehicle and simultaneously building a map without any kind of external intervention.

This work is a first step towards the final goal of performing SLAM for an AUV in a natural unaltered environment based on the information coming from an imaging sonar and using a Doppler Velocity Log (DVL) together with a compass for dead reckoning. The sensors used for this research were not specially chosen for this purpose. The

goal was to evaluate the possibility of using the sensor suite available on GARBI^{AUV} to perform SLAM. Solving the SLAM problem in natural environments is very challenging. When working under water using acoustic imaging sensors it is even more difficult. For this reason, it is not the goal of this early work to present an operational solution for the SLAM problem in the above mentioned conditions but to open a path towards this goal.

1.2 The GARBI^{AUV}

The GARBI platform was first conceived as a Remotely Operated Vehicle (ROV) designed with the aim of building an underwater vehicle using low cost materials, such as fiber-glass and epoxy resins. At the moment, an upgrading process is being carried out to transform this vehicle into an AUV for exploration in waters up to 100 meters in depth. The GARBI^{AUV} has a complete sensor suite including an imaging sonar, a DVL , a compass, 2 inclinometers, a pressure gauge, a temperature sensor, a DGPS unit and a colour camera. The control hardware is enclosed into two cylindrical hulls designed to withstand pressures of 11 atmospheres. Two additional cylinders containing the batteries are placed at the bottom, under the vehicle's gravity center, in order to ensure stability in both *pitch* and *roll* degrees of freedom. Its five thrusters will allow the GARBI^{AUV} to be operated in the remaining degrees of freedom (*surge*, *sway*, *heave* and *yaw*) and achieve a maximum speed of 3 knots. It weights 170 Kg approximately and its dimensions are 1.3 m in length, 0.9 m in height and a width of 0.7 m.

As can be seen in Figure 1.1 the remodeling is near to be completed. At the time of the work presented in this document the GARBI^{AUV} was still not operative so, the sensors had to be used independently of the vehicle. However, the GARBI^{AUV} deployment at sea is scheduled for July 2005, when it will be used for gathering an improved data-set for SLAM purposes.

Figure 1.1: GARBI^{AUV} updating process: a) Previous version. b) Updated version. c) First tests in a water tank.

1.3 Work outline

As introduced earlier, the final objective of this work is to develop a SLAM system to be used in underwater environments based on the information coming from the sensors suite of the GARBI^{AUV}. The first step towards this goal was the study of the related bibliography and the learning of the principal techniques involved in the SLAM methodology. (Chapters 2 and 3). Next, a mission was realized in a real natural environment to obtain a data-set for testing these techniques. The experiment consisted on the realization of a small trajectory while using an imaging sonar, a DVL and compass. The ground truth trajectory was obtained using a GPS unit. A detailed description on the equipment and the methodology applied to the obtention of this data-set is presented in Chapter 4. After, the studied techniques were implemented and tested in order to identify the principal issues affecting performance of this approach. Finally, a roadmap is proposed to face further experiments to achieve our goals (Chapters 5, 6 and 7).

1.4 Objectives

The objectives of this work are to:

1. Study the EKF feature-based solution to the SLAM problem.

2. Study the sensor suite available on GARBI^{AUV} for SLAM purposes.

3. Propose an experimental set-up for obtaining an underwater data-set for SLAM.

4. Study the difficulties related to a possible EKF feature-based implementation of SLAM for an AUV using an Imaging sonar for feature extraction and a DVL and compass for dead-reckoning.

5. Gain experience concerning the conditions in which the SLAM problem can be successfully faced with the sensor-suite mentioned above, and propose a roadmap to achieve it.

Chapter 2

State of the Art

2.1 Introduction

In many AUV applications, navigation is a critical issue. The positional error growth associated with dead reckoning based on Inertial Navigation Systems (INS) and/or DVLs make their use impractical for long term navigation. In order to avoid this problem, data from GPS 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 LBL or Ultra Short Baseline (USBL) (see Figure 2.1). Unfortunately, there are numerous missions in which these solutions are unworkable. The need of previous 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 from the environment [31]. Even so, its main drawback is that an up-to-date map of sufficient resolution will not be available on 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.

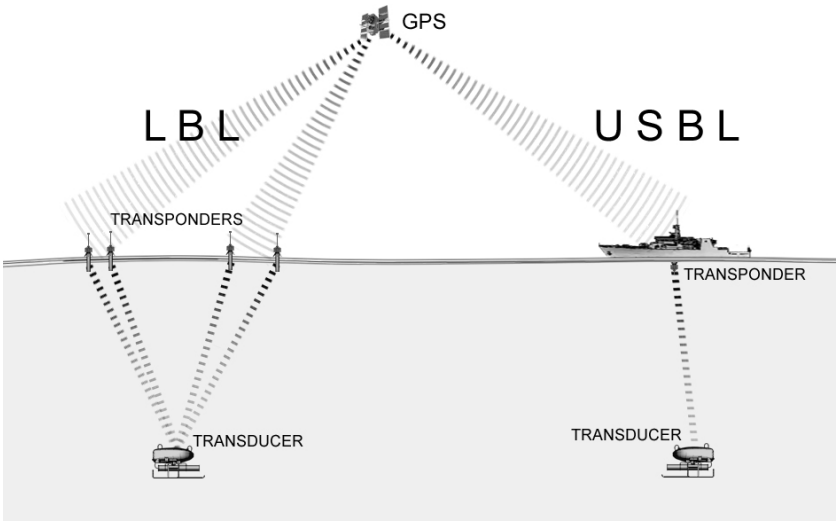


Figure 2.1: Typical configurations for AUV localization using LBL and USBL.

2.2 Simultaneous Localization and Mapping

Simultaneous Localization and Mapping (SLAM), also referred to as Concurrent Mapping and Localization (CML), is a technique created in the robotics community while wondering if an autonomous vehicle can start in an unknown location, without previous knowledge of the environment, and then incrementally build a map of this environment by extracting information from its sensors, while simultaneously using this map to compute the vehicle's location. The map and the estimates of the robot's location obtained from a successful SLAM system provide essential information on which high level tasks could be built. A practical solution to the SLAM problem is of inestimable value to achieving a truly autonomous mobile robot.

Over the last few decades, various approaches have been proposed to address this problem. It is possible to classify them depending on the representation method used for the map, which is a key point since it determines the kind of information explicitly expressed in the model. The most common representation methods are:

- **Occupancy grids.** The *occupancy grid* describes the vehicle's surroundings by dividing the space into multiple regular cells and assigning to each one a proba-

bilistic value that determines empty, unknown and occupied areas [11] [36].

- **Topological maps.** The spatial information is represented by graphs where the nodes symbolize characteristic places and the links correspond to the free spaces interconnecting them [24] [7].

- **Feature maps.** Feature based representations model the environment using a set of geometric primitives such as points, lines and planes previously extracted from raw sensor data [27] [6].

- **Scan chains.** The raw sensor data is directly aligned through a process of translation and rotation to find a maximum overlap with the world data contained in the map. It is suitable to use in situations where no simple geometric representation of the environment can be obtained, or is needed. Up to now, it has only been proven successful for laser [18].

Once a representation method is chosen, the main technical difficulty for performing SLAM is how to deal with uncertainty growth produced by the noisy sensor measurement [10]. It is essential to obtain a computationally efficient method capable of coping with the **uncertainty**, to represent it accurately and even to reduce it to obtain a precise map and, hence, a reliable vehicle localization. Other technical difficulties related to the SLAM problem are **feature extraction** and **data association** problems. Obtaining information from noisy sensor data and discerning whether the region currently observed is a newly explored territory or one that has been visited previously are key issues to ensure the convergence of a SLAM algorithm.

These three subjects (uncertainty, feature extraction and data association) are discussed in the following sections .

2.2.1 Uncertainty

Over the last few decades, various approaches have been proposed to face the problem of dealing with uncertainty. The work of Smith, Self and Cheeseman [43] in the mid-80s,

who introduced stochastic mapping, meant the first serious advance toward solving the SLAM problem. This method, based on the Kalman Filter (KF), described a representation of the spatial information using a variable-dimension state vector containing the estimates of relationships among features in the map along with their uncertainties. The KF provides an optimal linear recursive solution to SLAM when certain assumptions hold, such as perfect data association, linear motion and measurement models and Gaussian error models. This approach can also be extended to non-lineal problems through the Extended Kalman Filter (EKF) at the cost of losing the optimality.

Actually, feature based stochastic mapping using EKF is the most widely used method. Some remarkable examples of recent work include [9] using vision, [6] [17] [22] using laser sensors and [44] [25] with sonar data.

Nevertheless, in spite of its popularity, technics based on KFs suffer from the map-scaling problem: as the number of features stored in the state vector increases, the computational cost of updating the correlations between the state estimates increases quadratically.

Alternatively, other implementations such as Information Filters (IF) and its extended non-linear version, the Extended Information Filter (EIF), have been used in order to reduce this computational cost [47]. Like the KF, the IF 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 naturally sparse matrixes that offer a better computational efficiency.

Some other efficient strategies for SLAM with feature-based representations and Gaussian uncertainty models include postponement [23], decoupled stochastic mapping [25], the compressed filter [17], sequential map joining [44] and the constrained local submap filter [50].

Another important shortcoming of these kinds of filters is the limited capacity of Gaussian distributions to represent the uncertainty in many estimation problems. The use of non-parametric filters intends to improve this representation through the approxi-

mation of the uncertainty by a finite number of values. The most relevant nonparametric implementation is the particle filter [46] [34]. This uses a finite number of random sample states drawn from the estimate, called particles, to represent the uncertainty distribution. As the number of values increases, a better uncertainty description is obtained and the filter tends to converge at a better estimate. Luckily, it is possible to adapt the number of parameters to the suspected complexity of the estimate in order to obtain computationally efficient algorithms.

In contrast to feature based techniques, the Constant Pose Estimation (CPE) SLAM [30] [19] is a method that makes use of dense sensor data. It maintains a network of local constraints between robot positions, then the global map is produced through optimization. The main advantage of this method is that such a representation scales well as the area of the map grows because it generally represents only the local constraints.

2.2.2 Feature extraction

The representation election depends on the environment and the type of sensor used. This choice causes the data to be treated in a whole different manner. As seen previously, in a scan chains representation, dense sensor data (usually from laser scanners) is used directly as obtained from the sensor by aligning it to world data [29] [48].

Occupancy grids use a model of the sensor (commonly sonar or laser [15]) which provides information concerning empty and occupied volumes in a cone in front of the sensor. So, the data are transformed as probability profiles projected onto a rasterized map [37] [2].

In Feature-based methods, sensor data is processed in order to obtain well defined entities which can be detected and recognized repeatedly. These entities are called *features* and the process by which they are obtained *feature extraction*.

In these techniques the information coming from the sensors is manipulated to extract candidates which, in order to be a feasible feature, must accomplish the following criteria:

- **Discrimination.** Features must be noticeable or different from the rest of the

environment representation.

- **Invariance.** Features must be invariant in some aspects (scale, dimensions, orientation, etc) allowing recognition in future observations.

- **Uncertainty model.** Features must have a well defined uncertainty model in accordance with their characteristics.

The type of feature that can be obtained from data strongly depends on the sensor type and the environment in which the robot is moving. When working in structured areas, like indoor environments, features are usually parameterized as geometric objects. The most common primitives used are:

- **Point.** Usually obtained from beacons or objects that could easily be assimilated into a point such as a pole transversally scanned by a laser [16] or a sonar [51]. Alternatively, points could be obtained from the intersection of multiple range-only measurements as in [40] where only the distance from a beacon is measured, so the circumference of its area can be defined. If other measurements from the same beacon are available, the intersection of the circumferences can determine the exact position of the point landmark.

- **Line.** This is the most common feature when working in structured environments. Typically, a laser scanner obtains a cloud of points from which the line is estimated, using multiple methods such as: Split and Merge [42], Hough [21], RANSAC [13], among others. Although it is not as commonly used as laser sensors, sonar can also be used to extract line features [44].

- **Surface.** This is handled much like line extraction but extrapolated to 3D. Similar methods, such as RANSAC or Hough, can be used to extract planes from a point cloud (usually from laser scanners). However, when enough data is available for surface extraction, other methodologies such as scan matching are preferable.

Nevertheless, geometric features are difficult to obtain from natural environments, simply because outdoor terrain does not conform to any simple geometric model. Outdoor environments need a more general feature to describe the scene. A common approach is to use shapes or *blobs* which can contain additional feature characteristics to help in their recognition. These characteristics can define shape (size, center of mass, area, perimeter, etc.), aspect (color, texture, intensity, etc.) or other pertinent information descriptors (i.e. mean and variance from a cloud of points). An example using this type of general feature in natural environments can be found in [3], where underwater objects are extracted from sonar data and characterized by their area, perimeter, and radial signature.

A particular case is the features extracted from vision systems. These are normally based on extracting interest points characterized according to some descriptive criterion. Typically, the image features extraction and characterization efforts are based on the Moravec interest points [35], Harris corners [20] or SIFT descriptors [28].

2.2.3 Data association

Data association is the process of determining the correspondence between the measurement and the world representation. When a correspondence is found, it is incorporated into the system in order to improve the estimate. If a correspondence is not possible, the next step is to determine if there is any new information, and if there is, it must be included in the world representation or o must be considered spurious and therefore rejected. Correct data association is crucial because an incorrect assignment will cause location estimation methods such as the EKF to diverge.

The data association methodologies applied to a problem depends on the type of sensorial data used. Applications using raw sensor data, such as scan chains, perform data association by matching the observed scan with a previous map representation. These type of techniques can be classified into two categories: those which use a geometric representation of the environment and those which use a raw data representation. An example of the first category can be found in [8] where points of a scan are matched to

an *a priori* model consisting of line segments. The second category includes methods such as the use of a cross correlation function to find the maximum overlapping of two scans characterized by their histograms [48], or a solution in which a point-to-point correspondence is done to calculate the scan alignment [29]. A similar problem appears when matching occupancy grids. A method in which the matching of two grid representations is found by computing the sum of the products of their corresponding cells is presented in [37].

Alternatively to the matching techniques used in raw data applications, the data association problem in feature-based methods relies on finding the correspondences between features through the comparative analysis of their discriminant and invariant characteristics. Principally, this comparison is made on a basis of feature position and then other characteristics can be used to improve the results. This problem is frequently addressed with the classic Individual Compatibility Nearest Neighbor (ICNN) methodology [1]. The Individual Compatibility (IC) between an observation and a feature in the map can be determined using an innovation test that measures the Mahalanobis distance [32]. Then, from the features which satisfy the IC criterion, the one with the smallest distance is chosen. In spite of its conceptual simplicity and computational efficiency, the ICNN test has low reliability in detecting spurious matchings when the vehicle error grows with respect to sensor error. This happens mainly because it only considers individual compatibility between an observation and a feature, but ignores the fact that this pairing must be jointly compatible with other individual compatible pairings to form a consistent hypothesis. A much more restrictive algorithm that considers this aspect is the Joint Compatibility Branch and Bound (JCBB) [38]. This approach explores the space of correspondence using a Branch and Bound (BB) search algorithm to find the hypothesis containing the largest number of Jointly Compatible (JC) pairings.

2.3 Underwater SLAM

The problem of SLAM becomes particularly challenging in underwater environments due to sensing limitations and the challenging constrains imposed by the natural unstructured terrain. Extract information from sensors such as sonar or cameras is a difficult task and usually provide poor or ambiguous representations of the seabed. In order to simplify this issue, the problem of underwater SLAM had been addressed using some type of previously deployed sonar targets, so identifiable and stable features can be obtained [39][51]. An alternative is using transponders as features and enable the AUV to deduce its locations on the fly by means of range only measurements [40]. However, a solution not requiring any previous set-up is preferable. An approach like this requires a methodology capable to interpret natural objects from the seabed in order to use them as features. Using geometric primitives such as points and lines is not possible in an underwater environment because they does not fit adequately to natural terrains. In [26][45] sonar measurements are used to perform SLAM by extracting features from compact regions of strong bottom backscatter. These regions are identified by their center of mass position and other shape characteristics in order to solve the data association problem. On the other hand, vision systems can also be used to perform SLAM by extracting features from images of the seabed [12][14], or as seen in [49] to fuse its information with sonar data to obtain a more detailed description of the terrain. An alternative representation is introduced in [33], where Sum of Gaussian (SOG) distribution is used to build a feature map of landmarks to represent the environment.

Finally, there are some works related to the issues while operating large areas. In [25] the decoupled stochastic mapping is presented. This technique reduces the computational cost by dividing the environment into multiple overlapping submap regions. Other interesting approach can be found in [41], where a solution is proposed by using a constant-time algorithm.

Chapter 3

SLAM Algorithms

3.1 Introduction

The aim of this chapter is to briefly describe the theoretics involved in the practical work that will be presented in this report. Further details can be found in [4].

The chosen SLAM framework is the feature-based stochastic mapping first introduced in [43] using the non-linear version of the Kalman filter, the extended Kalman filter. This methodology consists of a kind of recursive predictor-corrector algorithm in which the actual state of the vehicle is estimated from a process model fed with the input data from dead reckoning sensors and then, this is corrected through the repeated observation of the relationship between the vehicle and the external features from the environment obtained from proprioceptive sensors.

The outline of this description follows the formulation as presented in [4] and describes the structure of the state representation, the SLAM algorithm split in the prediction and correction steps and a description of the two principal methods used to perform data association.

3.2 SLAM using an EKF

3.2.1 State representation

In a standard EKF-SLAM representation, all of the system's information is represented by the augmented state vector $\mathbf{x}^B$ with estimated mean $\hat{\mathbf{x}}^B$ and estimated error covari-

ance $\mathbf{P}^B$:

$$\hat{\mathbf{x}}^B = E\left[\mathbf{x}^B\right]$$

$$\mathbf{P}^B = E\left[\left(\mathbf{x}^B - \hat{\mathbf{x}}^B\right)\left(\mathbf{x}^B - \hat{\mathbf{x}}^B\right)^T\right]$$

where $\hat{\mathbf{x}}^B$ contains the position and attitude of the vehicle R and the features included in the map $F_1 \ldots F_n$, all relative to a base reference B:

$$\hat{\mathbf{x}}^B = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \hat{\mathbf{x}}^B_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \end{bmatrix}$$

$\hat{\mathbf{P}}^B$, the estimated covariance matrix of $\hat{\mathbf{x}}^B$, represents on its diagonal the estimated error variance of the elements in the state vector and in the off-diagonal elements their estimated covariance, which is a measure of the correlation between the elements:

$$\mathbf{P}^B = \begin{bmatrix} \mathbf{P}^B_R & \mathbf{P}^B_{RF_1} & \ldots & \mathbf{P}^B_{RF_n} \\ \mathbf{P}^B_{F_1R} & \mathbf{P}^B_{F_1} & \ldots & \mathbf{P}^B_{F_1F_n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}^B_{F_nR} & \mathbf{P}^B_{F_nF_1} & \ldots & \mathbf{P}^B_{F_n} \end{bmatrix}$$

Initialization

The selection of the base reference B to initialize the stochastic map is a key issue. Building a map relative to a fixed reference normally requires assigning an initial level of uncertainty to the vehicle's position. As stated in [39], this uncertainty is the lower bound which determines the limit value of the covariance of any object location estimate in the map. In practice, due to linearizations, the vehicle's uncertainty could drop below this initial value, producing inconsistency in the estimation [5].

An alternative is to select as base reference the vehicle's position at step 0, so $B = R_0$. The main advantage in choosing this reference is that it permits initializing the map with a perfect knowledge of the base location:

$$\hat{\mathbf{x}}^B_0 = \hat{\mathbf{x}}^B_{R_0} = 0; \quad \mathbf{P}^B_0 = \mathbf{P}^B_{R_0} = 0$$

This avoids future states of the vehicle's uncertainty reaching values below its initial settings, since negative values make no sense. So, the above-mentioned drawback can be circumvented. In addition, it is possible to obtain the position of any element in the state vector with respect to any other necessary reference base by applying the appropriate transformations [44].

State augmentation

In order to incrementally build a map, the SLAM algorithm needs to be able to include the new observed features in the state vector. In order to perform this state augmentation, the feature can be added to the state vector using a composition transformation (see Appendix A) between the vehicle position R and the observed feature E:

$$\hat{\mathbf{x}}^B = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \hat{\mathbf{x}}^B_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \end{bmatrix} \quad \Rightarrow \quad \hat{\mathbf{x}}^B_{aug} = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \hat{\mathbf{x}}^B_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \\ \hat{\mathbf{x}}^B_E \end{bmatrix} \quad where \quad \hat{\mathbf{x}}^B_E = \hat{\mathbf{x}}^B_R \oplus \hat{\mathbf{x}}^R_E$$

Then, it is necessary to update the estimated error covariance matrix:

$$\mathbf{P}^B_{aug} = \mathbf{F}_{aug} \mathbf{P}^B \mathbf{F}^T_{aug} + \mathbf{G}_{aug} \mathbf{P}^R_E \mathbf{G}^T_{aug} \tag{3.1}$$

with:

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \mathbf{I} \\ \mathbf{J}_{1\oplus} & 0 & \dots & 0 \end{bmatrix}$$

$$\mathbf{G}_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{J}_{2\oplus} \end{bmatrix}$$

where $\mathbf{J}_{1\oplus}$ and $\mathbf{J}_{2\oplus}$ are the Jacobians of the composition transformation $\oplus$ and $\mathbf{P}^R_E$ is the covariance of the vector representing the location of feature E with respect to the vehicle position R .

3.2.2 Prediction step

Vehicle model

The general form of a vehicle displacement can be written as:

$$\mathbf{x}_{R_k}^{R_{k-1}} = \hat{\mathbf{x}}_{R_k}^{R_{k-1}} + v_k\,; \quad p(v_k) \sim N(0, \mathbf{Q}_k)$$

where $\hat{\mathbf{x}}_{R_k}^{R_{k-1}}$ represents the movement of the vehicle from the position at step $k-1$ to the position at step k, and v_k is an additive zero-mean white Gaussian noise representing the unmodeled perturbations.

In order to get the estimated final position of the vehicle with respect to the base frame B the following composition transformation must be applied (see Appendix A):

$$\hat{\mathbf{x}}_{R_k}^{B} = \hat{\mathbf{x}}_{R_{k-1}}^{B} \oplus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \tag{3.2}$$

$$\mathbf{P}_{R_{k|k-1}}^{B} = \mathbf{J}_{1\oplus}\mathbf{P}_{R_{k-1}}^{B}\mathbf{J}_{1\oplus}^{T} + \mathbf{J}_{2\oplus}\mathbf{Q}_k\mathbf{J}_{2\oplus}^{T} \tag{3.3}$$

where $\mathbf{J}_{1\oplus}$ and $\mathbf{J}_{2\oplus}$ are the Jacobians of the composition transformation.

Feature model

Features are representations of the environment which must be invariant in time. The estimated position of a feature i with respect to the base reference B can be modeled as:

$$\hat{\mathbf{x}}_{F_{i,k}}^{B} = \hat{\mathbf{x}}_{F_{i,k-1}}^{B} = \hat{\mathbf{x}}_{F_i}^{B} \qquad i = 1 \ldots n \tag{3.4}$$

using an estimated covariance matrix that is also stationary:

$$\mathbf{P}_{F_{i,k}}^{B} = \mathbf{P}_{F_{i,k-1}}^{B} = \mathbf{P}_{F_i}^{B} \qquad i = 1 \ldots n \tag{3.5}$$

Complete model

According to equations 3.2 and 3.4, the augmented state transition model for the complete system may be written as:

$$\hat{\mathbf{x}}^{B}_{k|k-1} = \begin{bmatrix} \hat{\mathbf{x}}^{B}_{R_{k-1}} \oplus \hat{\mathbf{x}}^{R_{k-1}}_{R_k} \\ \hat{\mathbf{x}}^{B}_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^{B}_{F_n} \end{bmatrix}$$

And from equations 3.3 and 3.5 the estimated error covariance at step k may be defined as:

$$\mathbf{P}^{B}_{k|k-1} = \mathbf{F}_k \mathbf{P}^{B}_{k-1} \mathbf{F}^{T}_k + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}^{T}_k$$

where:

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{J}_{1\oplus} & 0 & \dots & 0 \\ 0 & \mathbf{I} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{I} \end{bmatrix}$$

$$\mathbf{G}_k = \begin{bmatrix} \mathbf{J}_{2\oplus} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

3.2.3 Correction step

Assuming that a hypothesis $\mathcal{H}_k$ which associates the measurements z_k with the features in the map is known (see Section 3.3), a correction of the state estimate can be done through the standard EKF update equations.

The first step is to obtain a prediction of the measurements using the sensor model defined by the nonlinear stochastic difference equation $h_{\mathcal{H}_k}$ and the information about the vehicle's and the feature's location from the state vector $\mathbf{x}^{B}_{k|k-1}$:

$$\mathbf{z}_k = h_{\mathcal{H}_k}(\mathbf{x}^{B}_{k|k-1}) + w_k; \quad p(w_k) \sim N(0, \mathbf{R}_k)$$

where w_k is an additive zero-mean white Gaussian noise representing the unmodeled perturbations inherent to any measurement. Next, with the innovation term $\nu_{\mathcal{H}_k}$, the

discrepancy between the measurements and their predictions can be determined. Its value and covariance are:

$$\nu_{\mathcal{H}_k} = \mathbf{z}_k - h_{\mathcal{H}_k}(\hat{\mathbf{x}}^B_{k|k-1})$$

$$\mathbf{S}_{\mathcal{H}_k} = \mathbf{H}_{\mathcal{H}_k} \mathbf{P}^B_{k|k-1} \mathbf{H}^T_{\mathcal{H}_k} + \mathbf{R}_k$$

where $\mathbf{H}_{\mathcal{H}_k}$ is the Jacobian of the nonlinear function $h_{\mathcal{H}_k}$

$$\mathbf{H}_{\mathcal{H}_k} = \left. \frac{\partial h_{\mathcal{H}_k}}{\partial \mathbf{x}^B_{k|k-1}} \right|_{\hat{\mathbf{x}}^B_{k|k-1}}$$

Then, the Kalman filter gain $\mathbf{K}_{\mathcal{H}_k}$ is obtained as:

$$\mathbf{K}_{\mathcal{H}_k} = \mathbf{P}^B_{k|k-1} \mathbf{H}^T_{\mathcal{H}_k} \mathbf{S}^{-1}_{\mathcal{H}_k}$$

Finally, the estimate of the system state vector and its covariance are improved adding the information from the measurements as follows:

$$\hat{\mathbf{x}}^B_k = \hat{\mathbf{x}}^B_{k|k-1} + \mathbf{K}_{\mathcal{H}_k} \nu_{\mathcal{H}_k}$$

$$\mathbf{P}^B_k = (\mathbf{I} - \mathbf{K}_{\mathcal{H}_k} \mathbf{H}_{\mathcal{H}_k}) \mathbf{P}^B_{k|k-1}$$

3.3 Data association

Data association consists of determining the possible matchings of each measurement with a feature in the map, resulting in the hypothesis:

$$\mathcal{H}_k = [\, j_1 \; j_2 \; \cdots \; j_m \,]$$

where j_i represents the association at step k of observation $\mathbf{z}_{k,i}$ from an environment feature E_i $(i = 1 \ldots m)$ with its corresponding map feature F_j $(j = 1 \ldots n)$. If $j_i = 0$, this indicates that no association with any existing feature in the map is possible.

As seen in Section 3.2.3, if a measurement $\mathbf{z}_{k,i}$ is correctly associated with a feature in the map F_j $(j = 1 \ldots n)$, then a correction of the state can be performed obtaining

a more precise map. If the observed feature does not have any possible association, it has to be added to the map as a new feature. This is called map building.

In the sections below, two popular methods of data association will be presented; the *Individual Compatibility Nearest Neighbor* and the *Joint compatibility Branch and Bound*.

3.3.1 Individual Compatibility Nearest Neighbor

The Individual Compatibility Nearest Neighbor (ICNN) is a simple method to establish the pairings according to the discrepancy between the predicted measurement and the actual sensor measurement.

Dropping the k subindex for clarity, the sensor observation model h_j is used to obtain the prediction of a single measurement $\mathbf{z}_i$ as the relationship of vehicle and feature location, both contained in the map state vector $\mathbf{x}^B_{k|k-1}$:

$$\mathbf{z}_i = h_j(\mathbf{x}^B_{k|k-1}) + w_i; \quad p(w_i) \sim N(0, \mathbf{R}_i)$$

where w_i is an additive zero-mean white Gaussian noise.

So the discrepancy between the prediction and the actual sensor observation is given as the innovation term ν_{ij}:

$$\nu_{ij} = \mathbf{z}_i - h_j(\hat{\mathbf{x}}^B_{k|k-1})$$

with covariance:

$$\mathbf{S}_{ij} = \mathbf{H}_j \mathbf{P}^B_{k|k-1} \mathbf{H}^T_j + \mathbf{R}_i$$

where:

$$\mathbf{H}_j = \left. \frac{\partial h_j}{\partial \mathbf{x}^B_{k|k-1}} \right|_{\hat{\mathbf{x}}^B_{k|k-1}}$$

A pairing is accepted if its individual compatibility (IC) determined by the innovation Mahalanobis distance D^2_{ij} [32] satisfies:

$$D_{ij}^2 = \nu_{ij}^T \mathbf{S}_{ij}^{-1} \nu_{ij} < \chi_{d,\alpha}^2 \tag{3.6}$$

where $d = \dim(h_j)$ and α is the desired confidence level.

If this test is applied to the predicted state, a subset of map features compatible with a measurement E_i can be determined.

The Nearest Neighbor (NN) selection criterion determines that among the features that satisfy (5.1), the one with the smallest Mahalanobis distance is chosen to establish hypothesis $\mathcal{H}_k$.

3.3.2 Constraints of ICNN

Although ICNN is a very simple and computationally efficient algorithm (it performs only nm tests), there is a high risk of obtaining inconsistent hypothesis since this test does not evaluate the joint compatibility of all measurements. To illustrate this limitation, consider the example in Figure 3.1.

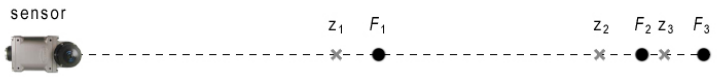


Figure 3.1: Monodimensional observation with three features

Assume that the sensor returns measurements $\hat{\mathbf{z}}_1$, $\hat{\mathbf{z}}_2$ and $\hat{\mathbf{z}}_3$, which correspond with features F_1, F_2 and F_3 in the map. Assume as well that measurement $\hat{\mathbf{z}}_1$ is individually compatible with F_1, $\hat{\mathbf{z}}_2$ with F_2 and $\hat{\mathbf{z}}_3$ with both F_2 and F_3 but its Mahalanobis distance to F_2 is smaller. According to IC, measurement $\hat{\mathbf{z}}_3$ has two possible pairings, but NN would determine that the correct hypothesis is $\{(\hat{\mathbf{z}}_1, F_1)(\hat{\mathbf{z}}_3, F_2)\}$, pairing $\hat{\mathbf{z}}_3$ with F_2 incorrectly. Considering the problem from a *jointly* compatible point of view, it could be deduced that the best hypothesis is $\{(\hat{\mathbf{z}}_1, F_1)(\hat{\mathbf{z}}_2, F_2)(\hat{\mathbf{z}}_3, F_3)\}$. In order to avoid this kind of error when using ICNN, two conditions must hold:

1. The vehicle error is smaller than the distance between the features, so it is unlikely that the two features pass the IC test for the same measurement.

2. Spuriousness is low enough so that it is unlikely that a spurious measurement will fall inside the acceptance region of some feature.

3.3.3 Joint Compatibility Branch and Bound

The Joint Compatibility Branch and Bound (JCBB) [38] is a method developed to solve the above mentioned constraints of the ICNN. In order to establish the consistency of the entire hypothesis $\mathcal{H}_k = [j_1 \;\; j_2 \;\; \cdots \;\; j_m]$, a test called Joint Compatibility (JC) is used. Analogous to IC, the JC test uses a joint observation model $h_{\mathcal{H}_k}$ to predict the measurements:

$$h_{\mathcal{H}_k}(\mathbf{x}^B_{k|k-1}) = \begin{bmatrix} h_{j_1}(\mathbf{x}^B_{k|k-1}) \\ \vdots \\ h_{j_m}(\mathbf{x}^B_{k|k-1}) \end{bmatrix}$$

With the joint innovation term $\nu_{\mathcal{H}_k}$ of value:

$$\nu_{\mathcal{H}_k} = \mathbf{z}_k - h_{\mathcal{H}_k}(\hat{\mathbf{x}}^B_{k|k-1})$$

and covariance:

$$\mathbf{S}_{\mathcal{H}_k} = \mathbf{H}_{\mathcal{H}_k} \mathbf{P}^B_{k|k-1} \mathbf{H}^T_{\mathcal{H}_k} + \mathbf{R}_k$$

where:

$$\mathbf{H}_{\mathcal{H}_k} = \begin{bmatrix} \mathbf{H}_{j_1} \\ \vdots \\ \mathbf{H}_{j_m} \end{bmatrix} \quad ; \quad \mathbf{H}_{j_m} = \left. \frac{\partial h_{j_m}}{\partial \mathbf{x}^B_{k|k-1}} \right|_{\hat{\mathbf{x}}^B_{k|k-1}}$$

measurements $\mathbf{z}_k$ can be considered jointly compatible with their corresponding features according to $\mathcal{H}_k$ if the Mahalanobis distance $D^2_{\mathcal{H}_k}$ satisfies:

$$D^2_{\mathcal{H}_k} = \nu^T_{\mathcal{H}_k} \mathbf{S}^{-1}_{\mathcal{H}_k} \nu_{\mathcal{H}_k} < \chi^2_{d,\alpha} \tag{3.7}$$

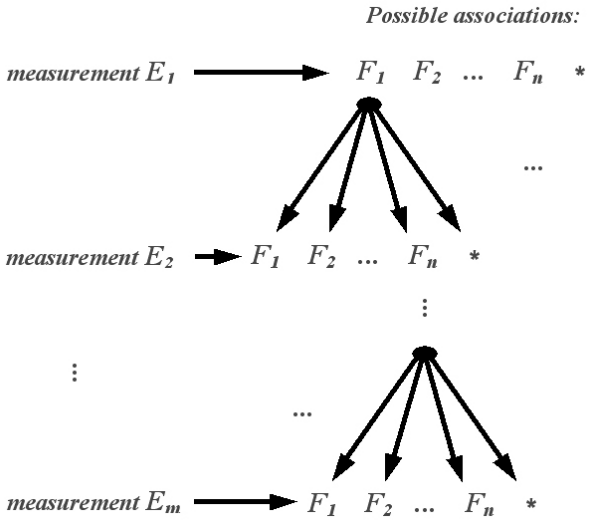


Figure 3.2: Interpretation tree

where $d = \dim(h_j)$ and α is the desired confidence level.

To obtain the best hypothesis we need an algorithm to search through all the possible hypotheses and find the one with the largest number of jointly compatible pairings. The space of observation-feature correspondences can be represented as an interpretation tree (see figure 3.2) of m measurement levels with $n+1$ branches corresponding to each possible pairing (including the possibility that association with any existing feature in the map is not available). The algorithm used to traverse the tree and search among the $(n+1)^m$ possible hypotheses the best one is the Joint Compatibility Branch and Bound data association algorithm. The JCBB is a particular type of back-tracking algorithm in which a cost function is used to prune the interpretation tree in order to reduce the search time drastically. For a more detailed description of JCBB see Algorithm 1.

Algorithm 1: Joint Compatibility Branch and Bound algorithm $JCBB(\mathcal{H})$ (reproduced from [4]):

if $i > m$ **then** {- *leaf node?*}
  if $pairings(\mathcal{H}) > pairings(Best)$ **then**
    $Best \leftarrow \mathcal{H}$
  end if
else
  for $j = 1$ *to* n **do**
    if $individual\_compatibility(i, j)$ **and then** $joint\_compatibility(\mathcal{H}, i, j)$**then**
      $JCBB([\mathcal{H}\ j], i + 1)$ {- *pairing* (E_i, F_j) *accepted*}
    end if
  end for
  if $pairings(\mathcal{H}) + m - i > pairings(Best)$ **then** {- *can do better?*}
    $JCBB([\mathcal{H}\ 0], i + 1)$ {- $*$ *node,* E_i *not paired*}
  end if
end if

Chapter 4

Experimental Set-up

4.1 Introduction

The work presented in this document is an attempt to perform SLAM in underwater environments extracting information from natural elements without the aid of external hardware such as artificial landmarks or beacons. In order to test these techniques, a data-set was setup during a mission in a real environment using the same sensor suite that will equip the GARBI^{AUV}. The sensor suite and the data-set acquisition procedure are described in this chapter.

4.2 Sensors

4.2.1 Miniking imaging sonar

The Tritech MiniKing is a small compact imaging sonar 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 sectors from a few degrees to full 360° scans. The characteristic fan-shaped beam (see Figure 4.1), 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 50 meters.

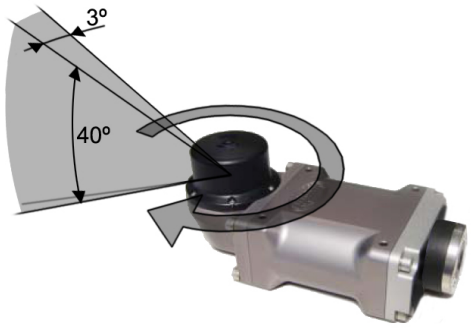


Figure 4.1: Beam dimensions of the Tritech Miniking imaging sonar.

Interpretation of sonar images

Understanding the scanning process which creates the sonar image is indispensable to interpreting the information obtained. Figure 4.2 shows the echo energy profile obtained from a fan-shaped beam intersecting with the bottom with an object present in the scene. The process starts when an acoustic signal or "ping" is sent from the transducer head and travels through the environment. When the wave collides with any object, part of the energy is returned, so if the time of flight is known, the distance to the object can be inferred and, in addition, by analyzing the returned sound intensity, some of the object's characteristics can also be obtained. All this information is given by the sensor as a vector of echo amplitude values whit each one corresponding to a predefined distance. In the example shown in Figure 4.2, the first return is obtained when the wave reaches the bottom. Then, as the acoustic signal advances and finds the object, a change in 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 intensity after an object detection. The length of this shadow gives information about the height of the object.

The next step is to form a complete image of the surroundings. If the transducer head is rotated at small regular angle increments and successive pings are emitted, then the received information can be analized to obtain a representation of the seabed (see 4.3).

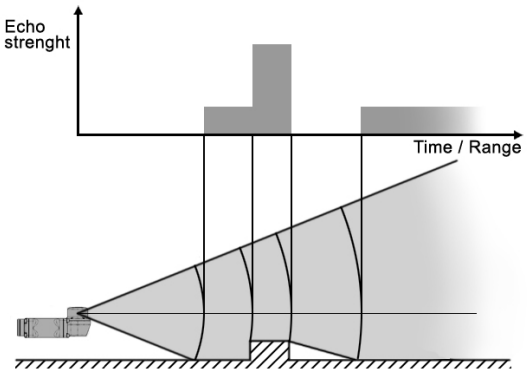


Figure 4.2: Echo strength vs distance from a fan-shaped sonar beam.

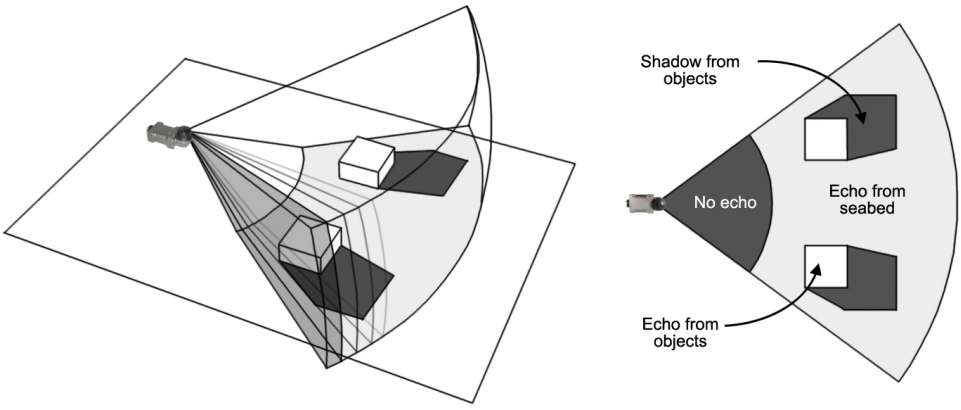


Figure 4.3: Generation of a sonar image.

Usually, the sonar image somewhat resembles an optical image of the same scene. In the example, the two boxes on the bottom can be recognized because they produce a higher amplitude return and, behind them, a shadow has been drawn as if the sonar head were a light source. However, a real sonar image is usually more difficult to interpret for many reasons. 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 making up the seabed will be determinant in order to obtain information from a sonar. Generally, rough objects are better sonar targets because they return the echoes in many different directions. On the other hand, smooth surfaces may give a very strong reflection in one

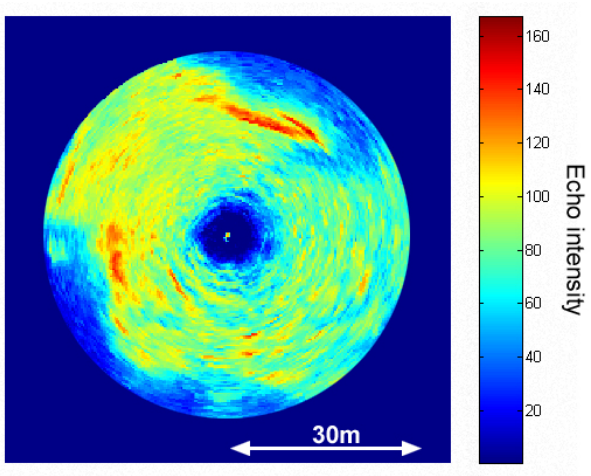


Figure 4.4: Sonar image from a natural environment.

particular direction, but almost none in any other.

The image in Figure 4.4 is a representation of real data obtained by the Miniking sonar in a shallow water trial. A 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 flat seabed and finally, the ones in blue represent zones from which no echo is returned, i.e. shadows. Notice that, as previously commented, shadows are usually found behind the high intensity zones.

4.2.2 Sontek Argonaut DVL

The SonTek Argonaut Doppler Velocity Log 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. 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 equally spaced at 120° relative azimuth angles. This beam geometry permits measuring velocities in 3D maintaining an optimal balance with total measurement range and near-boundary operation. The velocities can be obtained either in the BEAM coordinate system, in which the velocities are reported along the beams, or in the XYZ coordinate system,

Figure 4.5: The Sontek Argonaut DVL

in which velocity measurements are stored using a right-handed Cartesian coordinate system relative to the sensor. Alternatively, the Argonaut DVL is equipped with a compass/tilt sensor which allows transforming the velocities to an ENU (East-North-Up) coordinate system so, the data can be reported independently of instrument orientation.

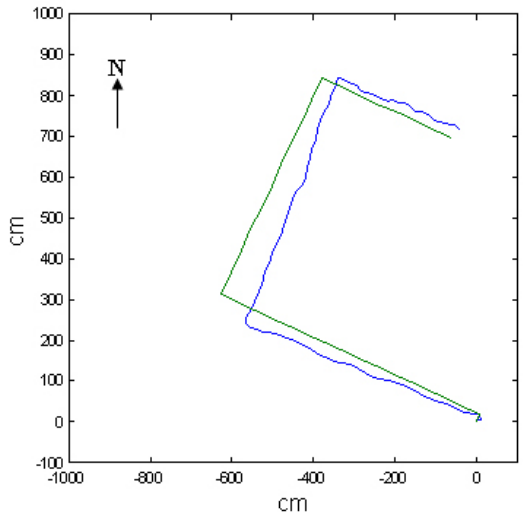


Figure 4.6: Trajectory obtained from Argonaut DVL velocity measurements.

In addition to the velocity measurements and the attitude data obtained from the compass/tilt sensor, the Argonaut DVL can also record the distance to the bottom for each of the three beams, estimate depth by means of a pressure sensor and even measure

water temperature for sound speed calculations. The Argonaut DVL is a versatile sensor which, together with its compact size, low power consumption and depth ratings of 200 m, makes it well suited for underwater vehicle positioning applications.

An example of a trajectory obtained using the velocity measurements from the DVL is shown in Figure 4.6. The trace in green represents the contour of a wharf where the DVL was deployed and the trace in blue is the sensed trajectory obtained by following the outline of the wharf over 17 meters.

4.2.3 Garmin eTrex Legend GPS

The Garmin eTrex Legend is a portable GPS unit designed to provide precise GPS positioning using correction data obtained from the Wide Area Augmentation System (WAAS). When receiving WAAS corrections the unit can achieve accuracy to less than 3 meters. However, when WAAS coverage is not available, the accuracy is to no less than 15 meters.

Figure 4.7: The Garmin eTrex Legend GPS.

Its most interesting characteristic is the possibility of saving position data while carrying out trajectories, so it can be recovered through a pc interface cable for later analysis. Although the resolution of the stored data is not very good, it is suitable for using in experimental situations where a small, cheap and easy to set-up system is

needed to obtain a ground truth for comparative purposes.

4.3 Data-set acquisition

The objective of our test mission was the obtention of a data-set to test different SLAM algorithms, especially those involving feature extraction and data association in natural environments. The experiments took place in September 2004 in a shallow water scenario along the shoreline near Colera, a small town in Girona, Spain. This location was chosen due to the variability of the seabed and the presence of multiple reefs and rocks for use as features for our SLAM approach. The experimental set-up consisted of a small boat with

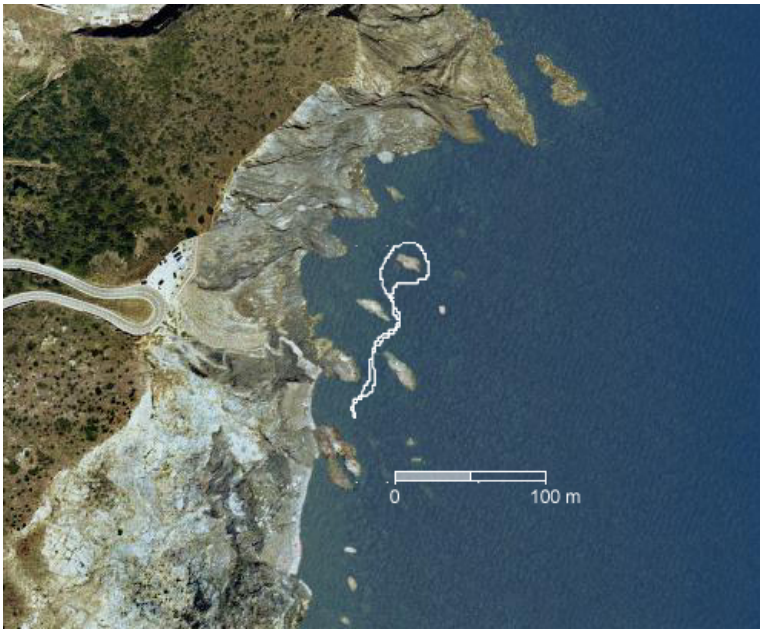


Figure 4.8: GPS trajectory traced on a satellite image

a metal structure held under its bow to hold the sensors, the Miniking imaging sonar and the Argonaut DVL. Simultaneously, the Garmin eTrex GPS unit was placed onboard to record the trajectories. During the mission, the sensor's structure was approximately 1 meter below the water's surface. The water depth in the area varied between 4 and 7 meters. The Miniking sonar was set to make continuous 360° scans, at a range of 30

meters, taking 100 intensity values each ping. The DVL provided heading and velocity estimates at its maximum rate, approximately each 0.6 seconds.

These experimental trials consisted of following trajectories for about 100 meters before returning to the starting point in zones with many rocks. Of the several trials, the one with the most interesting characteristics was chosen to be used as a data-set. The selected trajectory can be seen in Figure 4.8. As the GPS trace shows, the path crosses an initial zone making some small curves in order to pass near the rocks, then a small loop around a reef is made (about 30 meters) and finally, the trail leads back to the starting point so the initial zone was visited again.

Chapter 5

SLAM Implementation

5.1 Feature extraction

This section explains the process for feature extraction from the acoustic images obtained by the Miniking imaging sonar. The flowchart of the whole process is represented in Figure 5.1, then a detailed description of each step is given.

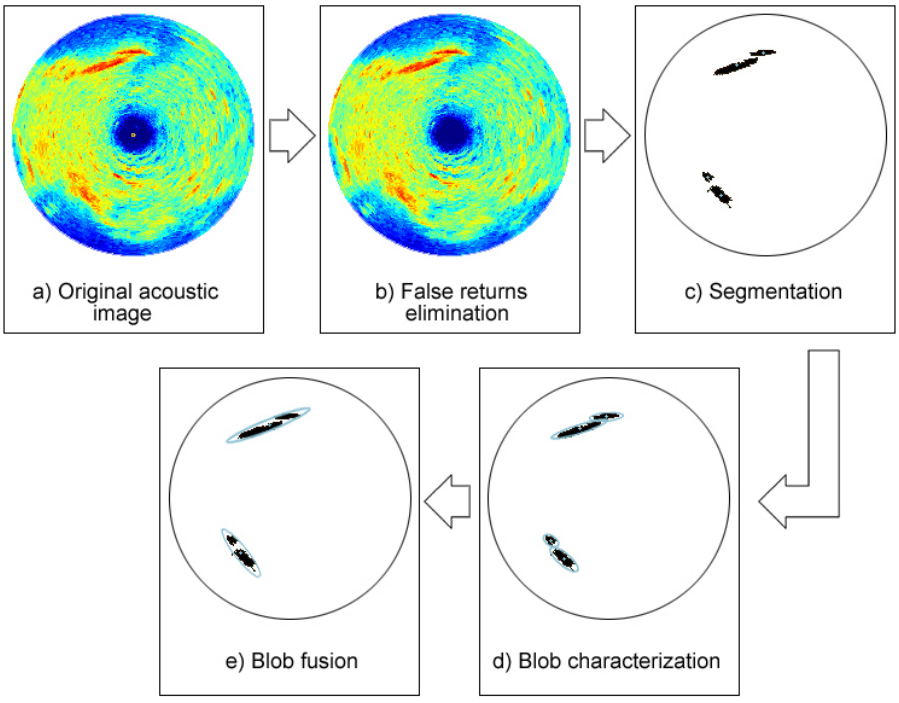


Figure 5.1: Flowchart of the feature extraction process.

5.1.1 Eliminating false returns

As seen previously, the imaging sonar scans an area by sequentially sending out pings at different bearings. When the transducer head receives the resulting echo, the information is quantized into discrete intensity values called "bins", each one corresponding to a predetermined distance. Figure 5.2 shows the intensity vs distance plot for a single ping in a shallow water environment.

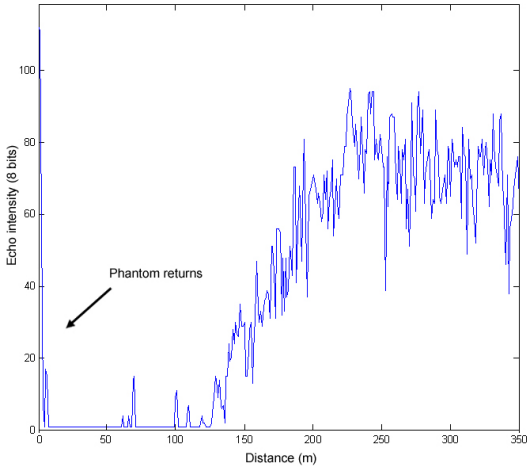


Figure 5.2: Single ping in shallow water.

Looking at the first bins, a confusing high amplitude returns is seen. These values near the sensor are false detections produced by a transient "ringing" of the transducer head following the sonar pulse emission. These false phantom returns must be filtered out before any further processing of the sonar data can be done. In order to do this suppression, the first bins of every ping are replaced with values representing a no echo return (see Figure 5.1 b).

5.1.2 Segmentation and blob characterization

The next step is to segment the high intensity areas from the acoustic image (Figure 5.1 b). First, an averaging filter is applied to diminish the effects of noise and to obtain a more uniform intensity profile. Next, the image is binarized by applying a

threshold. The measurements with an intensity value lower than 80% of the typical maximum value are discarded (the resulting image is shown in Figure 5.1 c). At this point, the high intensity blobs are obtained, but in order to discriminate the more persistent blobs, those with an area smaller than a predetermined value are also rejected. Finally, different properties of the blobs are obtained, such as a centroid, which will be used as a point feature in our SLAM system, and its covariance matrix. Knowing the covariance allows us to characterize the blob (which, in fact, is a cloud of points) with a covariance ellipse, making it easier to define properties such as orientation and scale (see Figure 5.1 d).

5.1.3 Fragmentation of blobs

Frequently, large blobs become fragmented into smaller ones (and vice versa) when they are re-observed in successive scans. Let us consider the case shown in Figure 5.3 where two consecutive observations of the same scene produce different blob representations. If features suffering from fragmentation are used, data association becomes more difficult and can produce erroneous pairings, resulting in inconsistency. On the other hand, discarding these features is not desirable since it means the loss of useful information.

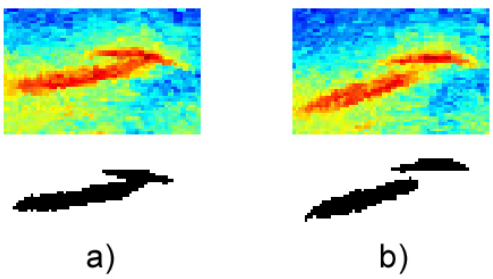


Figure 5.3: a) Feature obtained from a simple blob. b) The same feature split into two smaller blobs when re-observed.

This problem can be dealt with using a fusion procedure. From the image processing point of view, this kind of operation can be addressed using morphological operations (i.e. performing a dilation and an erosion). These operations allows closer regions in an

image to expand, until they fuse into a bigger one. The obtained result strongly depends on the type of mask chosen to perform the operations. However, the structural elements which produce better results are those which greatly alter the original shape of the regions. An alternative probabilistic approach can be used to solve this problem. After computing the covariance matrixes of the two blobs, a compatibility test (analogous to the one presented in 3.3.1) can be carried out in order to determine if they are close enough to be considered as a single feature.

Let $\mathbf{x}^B_{centr}$ be a vector containing the centroid coordinates of a pair of candidate blobs to be associated with the observation of a single feature, and let $\mathbf{P}^B_{centr}$ be its covariance. The limit case in making the association is that the two blobs are so close that they are coincident. This can be expressed by an ideal measurement equation without noise:

$$\mathbf{z}_{centr} = h(\mathbf{x}^B_{centr}) = 0$$

The discrepancy between the measured distance between the two blobs and its expected value can be defined by:

$$\nu_{centr} = \mathbf{z}_{centr} - h(\hat{\mathbf{x}}^B_{centr})$$

with the covariance obtained as:

$$\mathbf{S}_{centr} = \mathbf{H}\mathbf{P}^B_{centr}\mathbf{H}^T$$

where:

$$\mathbf{H} = \left.\frac{\partial h}{\partial \mathbf{x}^B_{centr}}\right|_{\hat{\mathbf{x}}^B_{centr}}$$

An association is accepted if its individual compatibility (IC) determined by the innovation Mahalanobis distance D^2_{centr} satisfies:

$$D^2_{centr} = \nu^T_{centr}\mathbf{S}^{-1}_{centr}\nu_{centr} < \chi^2_{d,\alpha} \tag{5.1}$$

where $d = \dim(h_j)$ and α is the desired confidence level.

If this association is positive, the two blobs are assigned to the same feature and consequently, its new characteristics (the centroid and the covariance) must be recalculated from the resulting set of their union. How a fragmented feature representation (d) is improved by joining the small blobs (e) can be observed in Figure 5.1. The result seems to be a better representation of the environment seen in Figure 5.1 a.

5.2 Algorithm implementation

Once the features are obtained the next step is to implement the SLAM system. In the following sections, the particularities of our approach using the formulation presented in Chapter 3 are presented.

5.2.1 State representation

As introduced in Section 3.2.1, all the system's information is represented by an augmented state vector containing the position and attitude of the vehicle and the features included in the map.

In our application, the vehicle state $\mathbf{x}_R^B$ is described by its coordinates, x_r and y_r, and its orientation, ϕ_r. All of these are represented in the two dimensional base reference B.

$$\mathbf{x}_R^B = \begin{bmatrix} x_r \\ y_r \\ \phi_r \end{bmatrix}$$

On the other hand, feature state $\mathbf{x}_{F_i}^B$ is represented only by its location coordinates. The situation of these point features correspond to the centroids obtained from the blobs in the acoustic images.

$$\mathbf{x}_{F_i}^B = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

5.2.2 Base reference initialization

As described in section 3.2.1, the initial vehicle position is selected as the base reference frame, without its orientation. As the only angular information available comes from

compass measurements, it is not possible to initialize any orientation-independent reference because all these measurements are relative to the geometric north. As shown in Figure 5.4, the chosen reference frame has its Y axis aligned with North, and requires an initial uncertainty value to its estimated orientation.

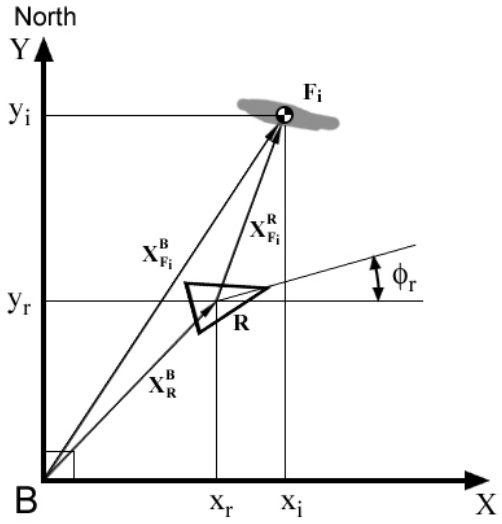


Figure 5.4: Vehicle and feature in the base coordinate frame.

5.2.3 Vehicle prediction model

The vehicle prediction model determines the evolution of the vehicle state vector. As previously described in Section 3.2.2, this spatial relationship could be defined as the compounding operation:

$$\mathbf{x}^B_{R_k} = \mathbf{x}^B_{R_{k-1}} \oplus \mathbf{x}^{R_{k-1}}_{R_k}$$

where $\mathbf{x}^{R_{k-1}}_{R_k}$ is the displacement produced between step $k-1$ and step k. In our implementation the estimated displacement is obtained by using the velocity measurements from the DVL sensor as follows:

$$\hat{\mathbf{x}}^{R_{k-1}}_{R_k} = \begin{bmatrix} V_x \Delta T \\ V_y \Delta T \\ 0 \end{bmatrix}$$

V_x and V_y are noisy velocity measurements from the Doppler sensor which, integrated through time, represents the position change in the vehicle coordinate frame. The angle increment is assumed to be small and modeled by the process noise.

The covariance matrix $\mathbf{Q}_k$ of the perturbations affecting the estimated displacement is defined as:

$$\mathbf{Q}_k = \begin{bmatrix} (\sigma_{V_x} \Delta T)^2 & 0 & 0 \\ 0 & (\sigma_{V_y} \Delta T)^2 & 0 \\ 0 & 0 & \sigma^2_{\phi_r} \end{bmatrix}$$

where σ_{V_x} and σ_{V_y} are the standard deviations of the noise affecting the velocity measurements and $\sigma^2_{\phi_r}$ is the covariance of a noise selected to emulate the possible changes in the orientation.

5.2.4 Feature measurement model

As introduced earlier, the chosen feature type is a point feature corresponding with the centroids of the detected blobs. As the measurements coming from the imaging sonar are polar, a change of reference is necessary to obtain a cartesian representation. Taking into consideration pitch and roll angles measured from the tilt sensors on the DVL and the range and bearing from the imaging sonar, the coordinates in a 2D vehicle-referenced base frame are obtained (see Figure 5.4). So, the feature's position in relationship to the vehicle can be represented by the estimated measurement vector $\hat{\mathbf{x}}^R_{F_i}$ and its associated covariance matrix $\mathbf{P}^R_{F_i}$. Notice that this covariance matrix is not related in any way to the one introduced in Section 5.1.2. The first represents the uncertainty of the centroid position produced by the noise affecting the sensor measurements, while the second is only used to represent blob characteristics for data association purposes (i. e. shape, principal axis, orientation...).

In order to incorporate newly observed features in the map, a composition transformation can be used to switch the feature from the vehicle reference R to the base reference B:

$$\hat{\mathbf{x}}^B = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \hat{\mathbf{x}}^B_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \end{bmatrix} \quad \Rightarrow \quad \hat{\mathbf{x}}^B_{aug} = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \hat{\mathbf{x}}^B_{F_1} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \\ \hat{\mathbf{x}}^B_{F_i} \end{bmatrix} \quad where \quad \hat{\mathbf{x}}^B_{F_i} = \hat{\mathbf{x}}^B_R \oplus \hat{\mathbf{x}}^R_{F_i}$$

$$\mathbf{P}^B_{aug} = \mathbf{F}_{aug}\mathbf{P}^B\mathbf{F}^T_{aug} + \mathbf{G}_{aug}\mathbf{P}^R_{F_i}\mathbf{G}^T_{aug} \tag{5.2}$$

where $\mathbf{F}_{aug}$ and $\mathbf{G}_{aug}$ are the as described in Section 3.2.1. In a similar manner, when a prediction of the measurement vector is necessary, it can be obtained from the vehicle $\hat{\mathbf{x}}^B_R$ and feature $\hat{\mathbf{x}}^B_{F_i}$ estimated state vectors, both present throughout the whole system state vector $\hat{\mathbf{x}}^B$:

$$\hat{\mathbf{x}}^R_{F_i} = (\ominus\hat{\mathbf{x}}^B_R) \oplus \hat{\mathbf{x}}^B_{F_i}$$

$$\mathbf{P}^B_{F_i} = \mathbf{J}_{1\oplus}(\mathbf{J}_{\ominus}\mathbf{P}^B_R\mathbf{J}^T_{\ominus})\mathbf{J}^T_{1\oplus} + \mathbf{J}_{2\oplus}\mathbf{P}^B_{F_i}\mathbf{J}^T_{2\oplus}$$

where $\mathbf{J}_{1\oplus}$, $\mathbf{J}_{2\oplus}$ and $\mathbf{J}_{\ominus}$ are the Jacobians of the respective transformations (see Appendix A).

5.3 Compass correction

Compass measurements allows periodical heading adjustments. These are made by merging the absolute angle measurement with the estimated orientation using the standard Kalman filter equations. In order to make any correction, we need a compass observation model:

$$\mathbf{z}_{comp,k} = h_{comp}(\mathbf{x}^B_{k|k-1}) + w_k; \quad p(w_k) \sim N(0, \mathbf{R}_{comp,k})$$

where w_k is a white process noise with zero-mean and normal probability distribution. Once a measure from the compass z_k and its prediction $h_k(\hat{\mathbf{x}}^B_{k|k-1})$ are available, the innovation term ν_k can be obtained. The value and covariance are:

$$\nu_{comp,k} = \mathbf{z}_{comp,k} - h_{comp,k}(\hat{\mathbf{x}}^B_{k|k-1})$$

$$\mathbf{S}_{comp,k} = \mathbf{H}_{comp,k}\mathbf{P}^B_{k|k-1}\mathbf{H}^T_{comp,k} + \mathbf{R}_{comp,k}$$

where:

$$\mathbf{H}_{comp,k} = \left.\frac{\partial h_{comp,k}}{\partial \mathbf{x}^B_{k|k-1}}\right|_{\hat{\mathbf{x}}^B_{k|k-1}}$$

Then a correction of the state is performed by means of the standard EKF correction equations:

$$\mathbf{K}_{comp,k} = \mathbf{P}^B_{k|k-1}\mathbf{H}^T_{comp,k}\mathbf{S}^{-1}_{comp,k}$$

$$\hat{\mathbf{x}}^B_k = \hat{\mathbf{x}}^B_{k|k-1} + \mathbf{K}_{comp,k}\nu_{comp,k}$$

$$\mathbf{P}^B_k = (\mathbf{I} - \mathbf{K}_{comp,k}\mathbf{H}_{comp,k})\mathbf{P}^B_{k|k-1}$$

5.4 Data association

The ICNN and the JCBB data association methods have been tested during this project. The ICNN method was implemented directly as described in Section 3.3. However, in order to use the JCBB, some aspects had to be taken into consideration. The JCBB algorithm, as a condition to taking advantage of its robustness, needs to associate a whole set of measured features jointly with those already in the map. But, in some applications, the sensor sequentially produces measurements from environment features so slowly that a noticeable vehicle displacement can be performed between them. This is the case when using the Miniking imaging sonar. So, to use the JCBB, it is necessary to let the system detect features for a determined period (i. e. a complete 360° scan around the vehicle), to accumulate them while coherently maintaining the uncertainty growth and finally, to use the complete set of observations to perform the JCBB and establish a correct association hypothesis.

In addition, to improve the data association performance and avoid false associations, other complementary tests can be executed. The presented ICNN and JCBB methods only consider the location of the elements to establish their compatibility, but in fact, additional feature characteristics can restrict the possible associations.

As introduced in Section 5.1.3, fragmented features can be grouped to recover a correct representation. However, this is not always enough to insure uniformity in the observations. Usually, their size or shape change due to the inherent imprecision of the acoustic images. Herein, some complementary tests have been made comparing the shape of the covariance ellipses from the blobs in order to deal with this kind of variability. These tests consisted of comparing the size of the ellipse's principal axis using the Ic criterion to determine if the observation pertains to a feature. Moreover, other characteristics, such as area, orientation or radial signature, are candidates to be used this way.

Chapter 6

Results

6.1 Introduction

This chapter presents the analysis of the SLAM techniques described in Chapter 5 applied to the data-set obtained in the experiment reported in Section 4.3. After a former post mission analysis of the gathered data, it was clear that it would be very difficult to close a loop with the collected data-set. Nevertheless, we decided to go forward for two main reasons. First, this is our initial work in this field and we wanted to get some in depth knowledge of the difficulties that could arise during the feature detection and data association problems. Second, it is our belief that a careful analysis of this challenging data-set would allow us to learn more about the conditions that must hold for performing an experiment that would lead us towards a successful SLAM application in a natural underwater environment using a low cost AUV.

In the following sections we identify the important issues that should be taken into account in order to achieve a successful SLAM. Moreover a compendium of recommendations are proposed in order to improve future experiments.

6.2 Data-set analysis

In Figure 6.1 the dead reckoning trajectory, obtained from the velocity and heading measurements, is compared with the ground truth trajectory represented by the GPS trace. As can be seen, the result does not follow the GPS path. Although both trajectories

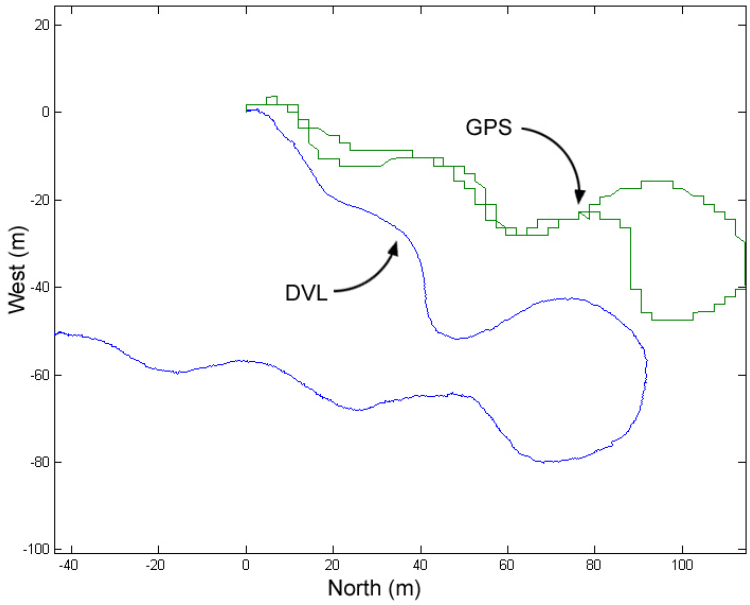


Figure 6.1: GPS vs dead reckoning representation.

have a similar length and shape, the dead reckoning trajectory is more open and does not close the loop. This error, mainly angular can be attributed to a bad calibration of the compass probably due to a magnetic disturbance of the structure on which the DVL was mounted. Unfortunately, this problem affected all the other trajectories as well so there was not any alternative data-set.

The imaging sonar results also have some deficiencies. In Figure 6.7 a sequence of scans corresponding to the whole trajectory is shown. Since the panoramic scans are gathered using a rotating sonar beam, and taking into account that the vehicle is moving during the scanning process, the sonar images are not perfectly circular as they would be in the case of static scans. Hence, in order to be able to interpret the images, the sonar beam reading must be plotted on an estimated trajectory. Note that, although in some cases the features can be easily identified and repetitively observed (take as example features 12, 13, 14 and 15) in other parts of the trajectory, it is difficult, even for a human, to distinguish features from noisy measurements (see scans S26, S27 and S28, where it is really difficult to find any correspondence).

Figure 6.2: Manually reconstructed interpretation of the feature map using the GPS trace

Figure 6.2 shows a speculative map of features placed on top of an ortophotomap of the area where the experiment was carried out. This map was set-up manually printing the detected features (see Figure 6.7) on the ground truth trajectory gathered with the GPS. The aim of this Figure is to show the detected features with respect to the real environment in which the robot operated. While looking at to this figure, it is worth noting that the map contains a small number of features. The measurements in the initial part of the trajectory have well-defined blobs which are repeatedly observed. However, the zones in which the greatest change is produced in the orientation (i.e. the final loop) coincide with zones with poor repeatability and low feature quality. It mainly depends on the angle of incidence of the acoustic signal and the distance of the observed features. The better observations are obtained when the detected object is located perpendicularly with respect to the sensor and on the other hand, detecting it at a long distance allows to obtain multiple observations of the same feature while approaching.

6.3 Algorithms analysis

6.3.1 Fusion of blobs

The probabilistic approach implemented to join blobs pertaining to the same feature obtains satisfactory results. In figure 6.7 the features extracted directly from the scans are shown in black ellipses, while the clustering established by the algorithm are represented by red ellipses. As can be seen in successive observations of the same feature (i. e. features 1, 3, 5, 7 and 9 or features 2, 4, 6, 8 and 10) the fusion algorithm is able to identify blob clusters of similar size and characteristics as the corresponding observations of the same unfragmented feature. As a result, more consistent features which can be correctly associated are obtained. However, there are a few cases the algorithm seems to fail (see features 35, 38 and 39).

6.3.2 SLAM using ICNN

In Figure 6.3 the resulting trajectory using ICNN can be observed. As can be appreciated, the trajectory has not improved with respect to the trajectory computed through dead reckoning. This is principally because of two reasons: the erroneous compass calibration and the scarcity of information obtained from the features. The incorrect angular measurements affected the data association since it propagates to the feature position. An uncalibrated compass presents a nonlinear polynomic relationship between the real heading and the measurement instead of presenting a linear relationship. This means that feature observed from the same robot position but with different headings will be mapped to different absolute map positions since the compass error is a nonlinear function of the orientation. Hence, this is a very important problem for the data association. In addition, as the compass gives absolute heading measurements, the small corrections in the orientation obtained through feature observation are rapidly corrupted by the erroneous sensor readings. As can be seen in the histogram represented in Figure 6.4, from 85 feature observations extracted from the acoustic images, only 23 observations where successfully associated with features in the map. This is really a small number

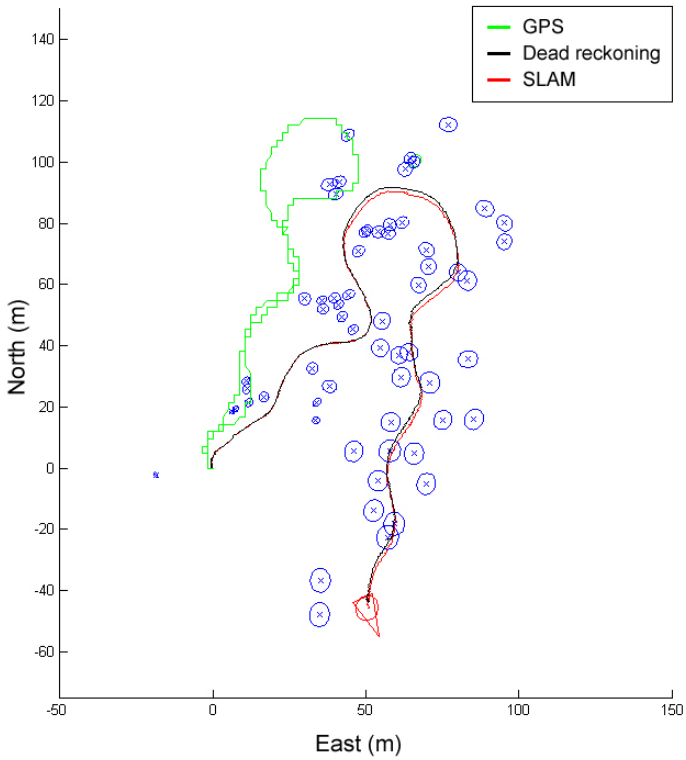


Figure 6.3: Resulting map using IC.

compared with the 1086 absolute orientation measurements obtained from the compass, hence its effect is minimal.

Observing the uncertainty ellipses drawn in Figure 6.3, especially the vehicle uncertainty, it can be argued that an overoptimistic error model was used since the final real position of the vehicle does not fall into the estimated robot ellipse. This fact suggests that a more pessimistic error model should be used. In the context of this work, a more pessimistic error model was also tested without getting better results. Increasing the uncertainty in the features generates more associations but also increases the number of false matchings. Moreover, if the compass uncertainty grows to include the magnetic perturbation, its weight in the correction step is so low that it is hardly taken into account provoking the vehicle follow an almost straight trajectory.

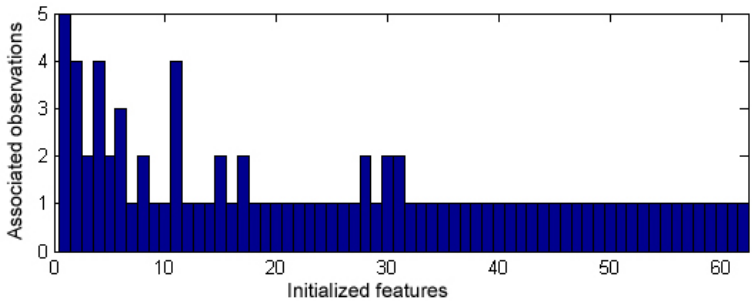


Figure 6.4: Feature association histogram using IC

6.3.3 SLAM using JCBB

In Figure 6.5 the same experiment is reproduced using the JCBB data association algorithm. The results are similar to the above using the ICNN. As introduced in Section 5.4, the JCBB method obtains a more robust solution than other methods such as IC by searching through a whole set of observations for a compatible hypothesis. The greater the set, the smaller the probability of a spurious observation being jointly compatible with the rest of the pairings in a hypothesis. So, in order to obtain advantages using this method, it is necessary to get as many observations as possible for each step. This is not the situation in our experiment. Returning to Figure 6.7, note that the maximum number of features in a single 360° scan is 5. However, the average number of observations is 2 or 3. Hence, there is no significant advantage in using the JCBB. It is also worth noting (Figure 6.6) that the number of resulting associations is similar to that obtained using IC.

6.4 Issues

From the previous analysis a compendium of the principal issues that require further attention can be obtained:

- **Compass calibration.-** Poor compass calibration is the principal source of problems in our approach. It affects our system at two different levels: giving erroneous

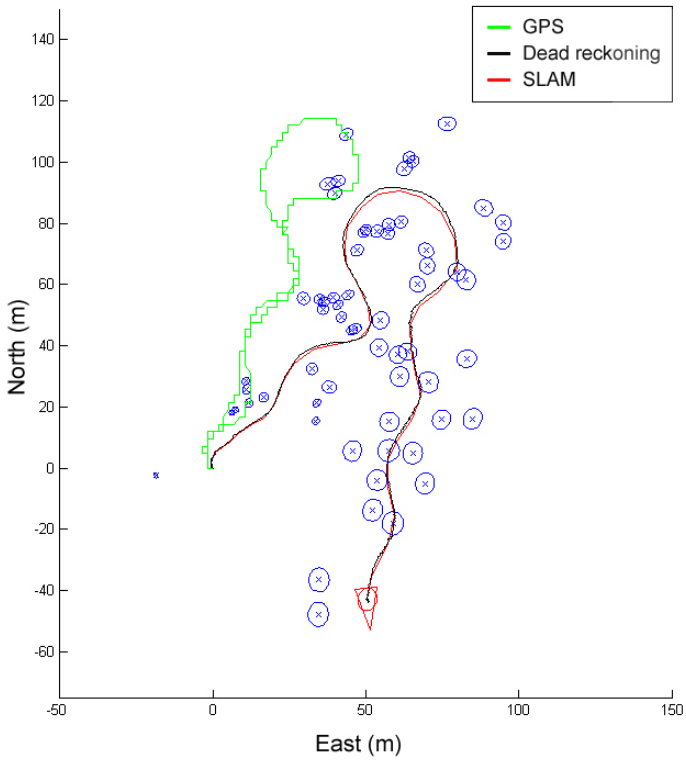


Figure 6.5: Resulting map using JC.

information to the process which estimates the heading of the vehicle and causing deformations in the observed blobs which makes further data association difficult.

- **Shortage of features.-** There are too few features observed in the trajectory, especially in zones where the vehicle passed very close to the rocks (i.e. the final loop around a rock in Figure 6.2). It seems that objects are observed better when they are at a certain distance. The JCBB will offer better results if more observations are available.

- **Low feature repeatability.-** If we look at the first scans in Figure 6.7, a good feature repeatability can be observed. However, as the vehicle increases velocity, the frequency of observation for any given feature decreases. This happens because the velocity of the vehicle is too high in comparison with the necessary time for the imaging sonar to perform a complete scan. A high feature repeatability is

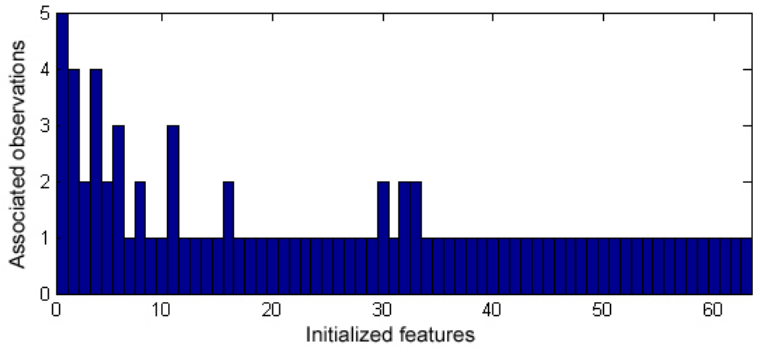


Figure 6.6: Feature association histogram using JC.

necessary to obtain a good set of inputs which will allow the SLAM system to obtain an accurate map and thus to converge.

- **Blob variability.-** Blob fragmentation seems to be correctly dealt with the proposed method. However, the data association method needs to be improved in order to deal with possible failures in the blob fusion methodology.

6.5 Proposals for future experiments

The following is a set of defined proposals calculated to address the mentioned issues:

- **Compass calibration.-** An accurate calibration must be achieved before any experimental trials. In addition, a calibration test can be carried out to ensure the quality of the obtained data. If it is not possible to get a correct measurement, using some type of complementary sensor such an INS could be taken into consideration.

- **Shortage of features.-** The trajectory must be planned keeping in mind the possible factors that can limit our visibility. It is necessary to avoid navigating too near the rocks, so the imaging sonar can obtain better observations. Also, it is necessary to avoid complex trajectories while in zones with few feature observations. Finally, manipulating the configuration of the imaging sonar can allow us

to scan larger areas by setting its range to the maximum (50 meters), augmenting the possibility of finding more features.

- **Low feature repeatability.-** One region should be scanned several times in order to obtain multiple observations of a feature. This can be achieved in two different ways: slowing down the vehicle or speeding up the scanning period of the imaging sonar. If the sensor is set up to obtain fewer measurements per ping, its processing velocity will be reduced and thus its scanning velocity will be increased, however, its resolution will get worse. In order to maximize the number of features and their repeatability, diverse tests must be carried out to determine an optimal sensor configuration.

- **Blob variability.-** Data association must be improved with additional procedures to allow the manipulation of this kind of variable feature. Data association algorithms which take into consideration additional feature characteristics (shape, orientation, intensity, perimeter, etc.) can help. However, the use of alternative feature type should not be discarded. One possible candidate can be some kind of representation based on splines to trace the contours of the detected reefs. Another option is to consider using some featureless representation such probabilistic grids.

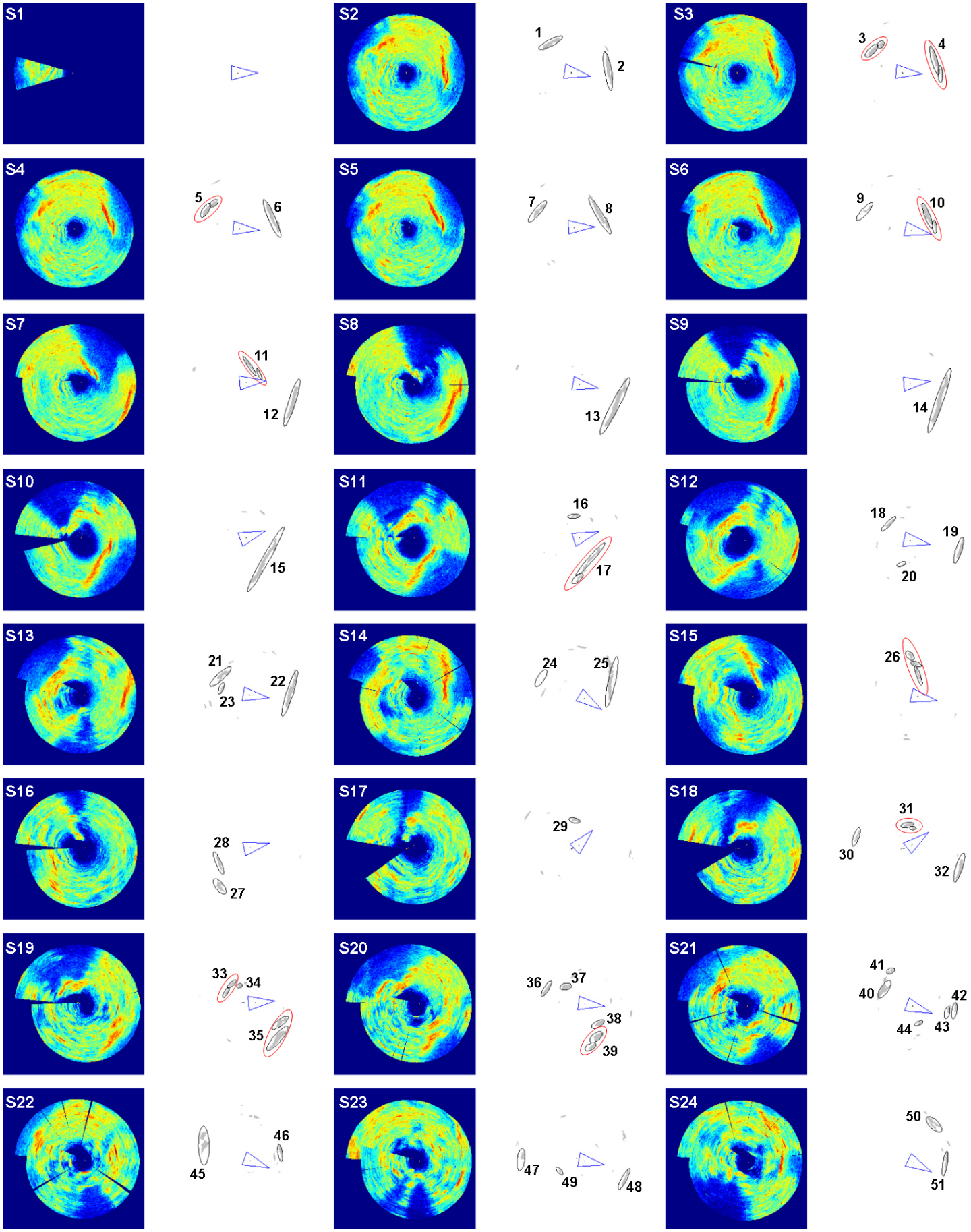
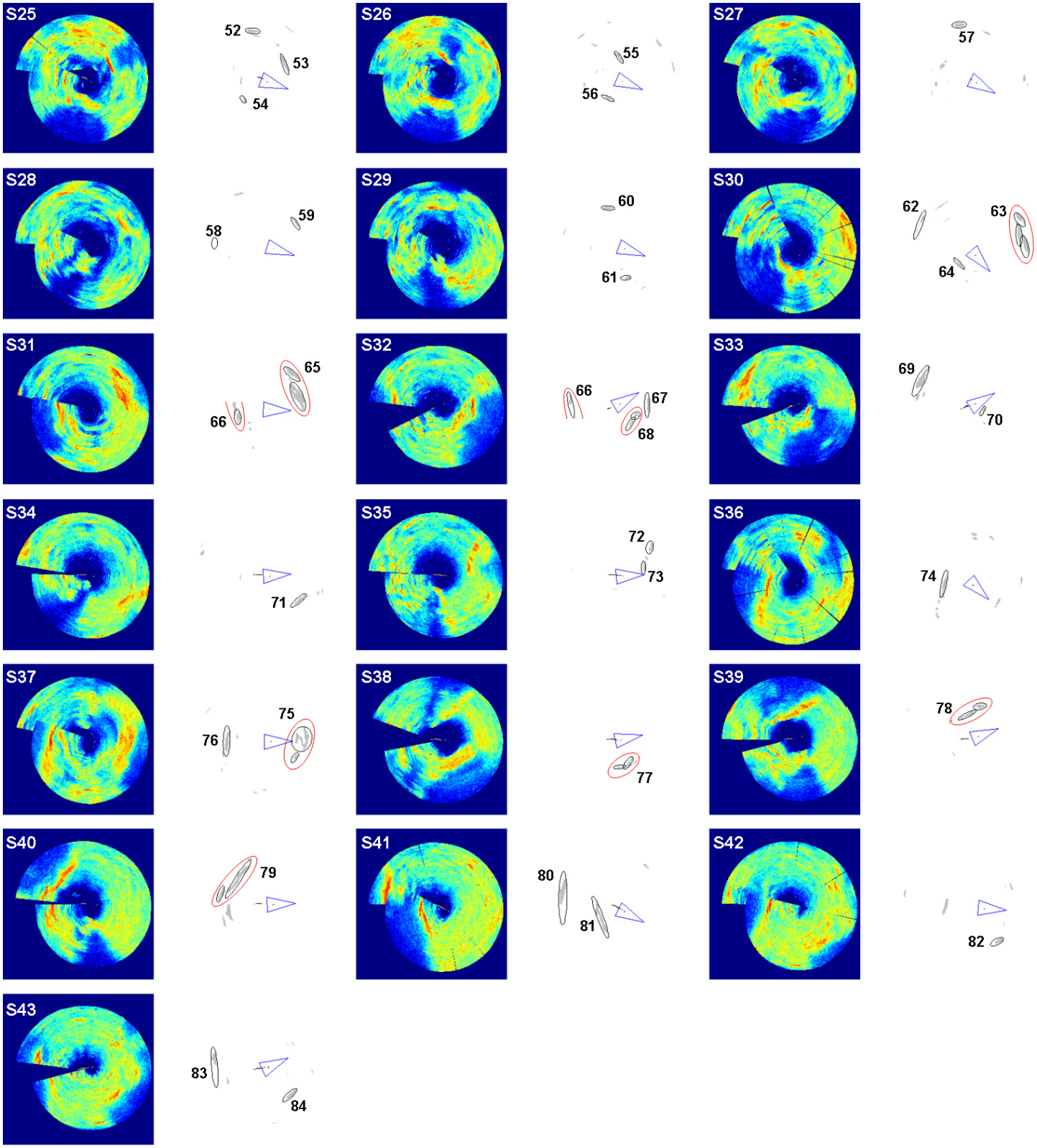


Figure 6.7: Sequence of imaging sonar scans for the whole trajectory.



Sequence of imaging sonar scans for the whole trajectory continued.

Chapter 7

Thesis Proposal

The objective of the proposed research is to develop a localization system for the GARBI^{AUV} using SLAM techniques. As the localization must be done in natural underwater environments, this work will pay special attention to the way the vehicle senses its surroundings using the available sensor suite. This study will affect two principal aspects of the SLAM methodology: the feature extraction and the data association problem. Our intention is to analyze the information gathered by an imaging sonar to determine different possible representations for feature extraction. The feature type that best fulfills the criterions of invariancy and distinctiveness will be chosen to implement the SLAM algorithm using the well known EKF formulation. Once the system has been properly verified through simple experimets, such as closing small loops, more complex scenarios will be faced using state-of-the-art formulation to deal with the computational demands of mapping large environments (i. e. the sub-map building approach presented in [44]).

7.1 Planning

A detailed schedule of the work to be carried out in this thesis is presented in Figure 7.1. Until now, the work has focused on gaining knowledge about the SLAM techniques and the sensors to be used. This knowledge has been obtained using a practical approach based on an experimental data-set acquired last year. The next step is to gather a new

data-set free of the errors defined in previous sections. In order to obtain a data-set of sufficient quality, the sensor suite will be properly calibrated and tested before the mission. Also, the GARBI^{AUV} has being updated since the beginning of the thesis. As the remodeling is near completion, it is expected that the sensor suite will be integrated in time to use the GARBI^{AUV} in the new mission. This experiment will consist of navigating along various paths with different characteristics. Our intention is to perform experiments firstly in a natural environment then complemented with artificial beacons, and then to evolve the complexity until we can perform a small loop in a natural, unmodified environment. Next, this data-set will be studied to propose a suitable feature representation in order to achieve a functional SLAM system. Once we have a working SLAM, a new data-set will be obtained so we can confront a more challenging situation for testing and improving the performance of the SLAM system. This mission will reproduce more complex trajectories simulating real-life applications, such as those involving survey trajectories for a bathymetric reconstruction. This will probably lead to an alternative formulation to deal with the computational growth involving large maps.

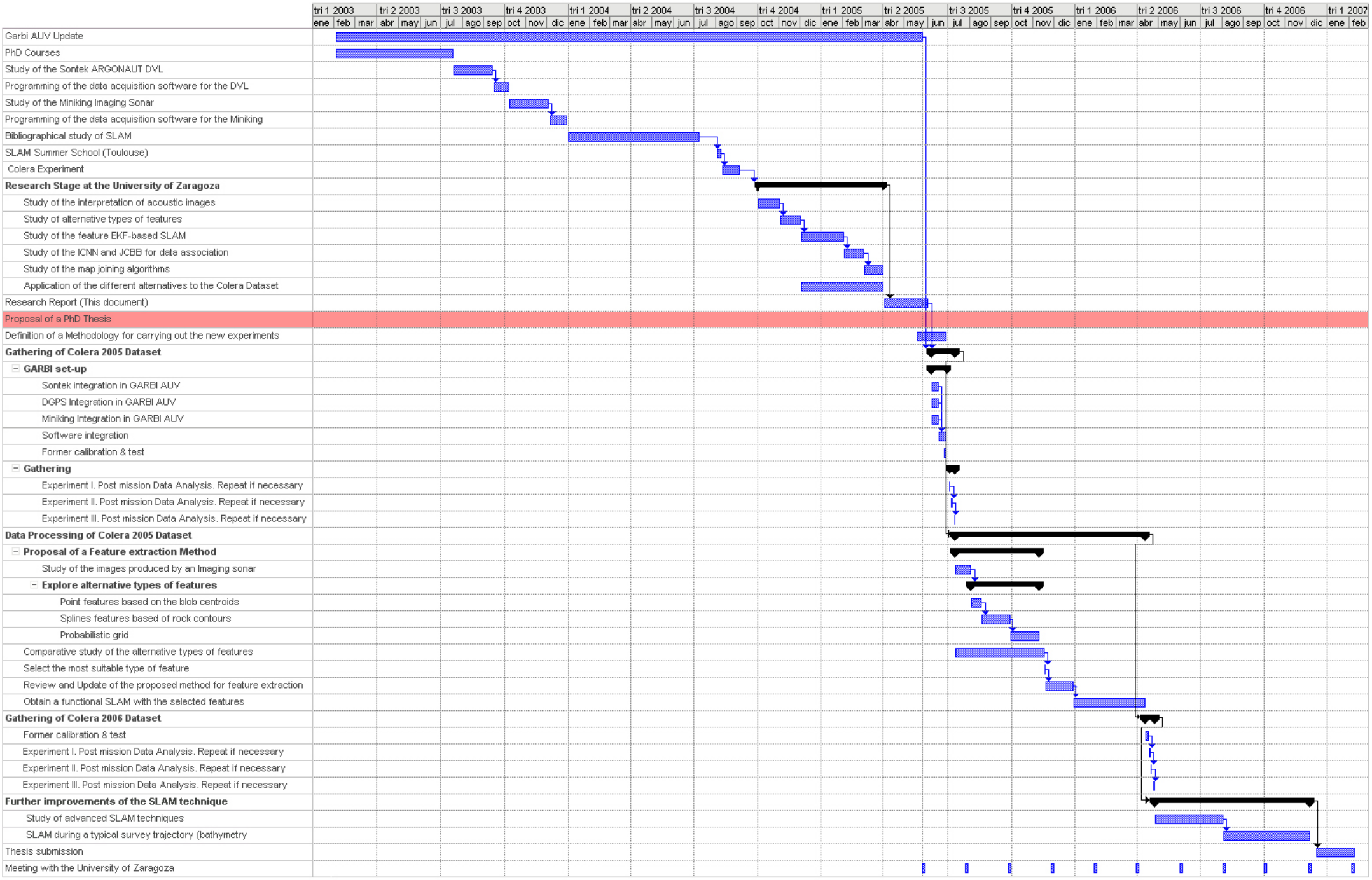


Figure 7.1: Schedule for the accomplishment of the proposed thesis.

Appendix A

Transformations in 2D

In [43] were presented two operations representing the most frequently encountered spatial relationships in stochastic mapping applications. These are the inversion and composition transformations, represented by the operators $\ominus$ and $\oplus$:

$$\mathbf{x}_C^A = \mathbf{x}_B^A \oplus \mathbf{x}_C^B$$

$$\mathbf{x}_C^A = \ominus \mathbf{x}_A^C$$

A.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 transformation:

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

The Jacobian of the inversion transformation is:

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

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

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

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

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

With compounded covariance as:

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

A.2 Composition

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 composition transformation:

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

The Jacobians of the composition transformation are:

$$\mathbf{J}_{1\oplus}\{\mathbf{x}_{B}^{A}, \mathbf{x}_{C}^{B}\} = \begin{bmatrix} 1 & 0 & -x_2 \sin\phi_1 - y_2 \cos\phi_1 \\ 0 & 1 & x_2 \cos\phi_1 - y_2 \sin\phi_1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{J}_{2\oplus}\{\mathbf{x}_{B}^{A}, \mathbf{x}_{C}^{B}\} = \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 & 0 \\ \sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

So, given the estimated mean and covariance of the spatial transformations:

$$\hat{\mathbf{x}}_{B}^{A} \; ; \; \hat{\mathbf{P}}_{B}^{A} \qquad \hat{\mathbf{x}}_{C}^{B} \; ; \; \hat{\mathbf{P}}_{C}^{B}$$

The estimated location of C relative to A can be described as the composition transformation:

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

With compounded covariance as:

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

A.3 Point features

Given the location of a 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 composition transformation:

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

The Jacobians of this transformation are:

$$\mathbf{J}_{1\oplus}\{\mathbf{x}_{B}^{A}, \mathbf{x}_{P}^{B}\} = \begin{bmatrix} 1 & 0 & -x_2\sin\phi_1 - y_2\cos\phi_1 \\ 0 & 1 & x_2\cos\phi_1 - y_2\sin\phi_1 \end{bmatrix}$$

$$\mathbf{J}_{2\oplus}\{\mathbf{x}_{B}^{A}, \mathbf{x}_{P}^{B}\} = \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 \\ \sin\phi_1 & \cos\phi_1 \end{bmatrix}$$

Again, given the estimated mean and covariance of the spatial transformations:

$$\hat{\mathbf{x}}_{B}^{A}\ ;\ \hat{\mathbf{P}}_{B}^{A} \qquad \hat{\mathbf{x}}_{P}^{B}\ ;\ \hat{\mathbf{P}}_{P}^{B}$$

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:

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

Bibliography

[1] Y. Bar-Shalom and T.E. Fortman. *Tracking and Data Association*. Academic Press, Boston, 1988.

[2] J. Borenstein and Y. Koren. Histogramic in-motion mapping for mobile robot obstacle avoidance. *IEEE Transactions on Robotics and Automation*, 7(4):535–539, August 1991.

[3] R. N. Carpenter. Concurrent mapping and localization with FLS. In *Proceedings of the Workshop on Autonomous Underwater Vehicles*, pages 133–148, Cambridge, MA, USA, 1998.

[4] J.A. Castellanos, J. Neira, and J.D.Tardós. *Map Building and SLAM Algorithms. In Autonomous Mobile Robots: Sensing, Control, Decision-Making, and Applications. Series in Control Engineering*. S.S. Ge and F.L. Lewis (Eds), Marcel Dekker, To appear.

[5] J.A. Castellanos, J. Neira, and J.D. Tardós. Limits to the consistency of EKF-based SLAM. In *Proocedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, July 2004.

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

[7] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. *IEEE Transactions on Robotic and Automation*, 17(2):125–137, April 2001.

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

[9] A.J. Davison. *Mobile Robot Navigation Using Active Vision*. PhD thesis, University of Oxford, 1998.

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

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

[12] R. Eustice, O. Pizarro, and H. Singh. Visually augmented navigation in an unstructured environment using a delayed state history. In *Proceedings ot the IEEE International Conference on Robotics and Automation*, New Orleans, USA, April–May 2004.

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

[14] R. Garcia, J. Puig, P. Ridao, and X. Cufi. Augmented state kalman filtering for auv navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4010–4015, Washington, 2002.

[15] G. Grisetti, C. Stachniss, and W. Burgard. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2443–2448, Barcelona, Spain, 2005.

[16] J. Guivant, E. Nebot, and H.F. Durrant-Whyte. Simultaneous localization and map building using natural features in outdoor environments. In *Proceedings of the IAS-6 Intelligent Autonomous Systems*, pages 581–588, Italy, 25-28 July 2000.

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

[18] J.-S. Gutmann and C. Schlegel. AMOS: comparison of scan matching approaches for self-localization in indoor environments. In *Proceedings of the First Euromicro Workshop on Advanced Mobile Robot*, pages 61–67, October 1996.

[19] J.S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proceedings of International Symposium on Computational Intelligence in Robotics and Automation, CIRA '99*, pages 318–325, November 1999.

[20] C.G. Harris and M.J. Stephens. A combined corner and edge detector. In *Proceedings of Forth Alvey Vision Conference*, pages 147–151, 1988.

[21] P.V.C. Hough. Method and means for recognizing complex patterns. U.S. Patent 3,069,654, December 18 1962.

[22] P. Jensfelt. *Approaches to Mobile Robot Localization in Indoor Environments*. PhD thesis, Signal, Sensors and Systems (S3), Royal Institute of Technology, SE-100 44 Stockholm, Sweden, 2001.

[23] J. Knight, A. Davison, and I. Reid. Towards constant time SLAM using postponement. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 406–412, October–November 2001.

[24] B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.

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

[26] J.J. Leonard, R.N. Carpenter, and H.J.S. Feder. Stochastic mapping using forward look sonar. *Robotica*, 19:341, 2001.

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

[28] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 1997.

[29] F. Lu. *Shape Registration using Optimization for Mobile Robot Navigation*. PhD thesis, University of Toronto, 1995.

[30] F. Lu and E. Milios. *Autonomous Robots*, volume 4, chapter Globally consistent range scan alignment for environment mapping, pages 333–349. Kluwer Academic Publishers, 1997.

[31] L. Lucido, J. Opderbecke, V. Rigaud, R. Deriche, and Z. Zhang. A terrain referenced underwater positioning using sonar bathymetric profiles and multiscale analysis. In *Proceedings OCEANS 96 MTS-IEE*, Fort Lauderdale, Florida (US), September 1996.

[32] P.C. Mahalanobis. On the generalized distance in statistics. In *Proc. Nat. Inst. Sci. India*, 1936.

[33] S. Majumder. *Sensor Fusion and Feature Based Navigation for Subsea Robots*. PhD thesis, Australian Centre for Field Robotics. The University of Sydney, August 2001.

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

[35] H. Moravec. Towards automatic visual obstacle avoidance. In *Proceedings of the 5th International Joint Conference on Articial Intelligence*, page 584, August 1977.

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

[37] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proceedings IEEE International Conference on Robotics and Automation.*, pages 116–121, March 1985.

[38] J. Neira and J.D. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on robotics and automation*, 17(6):890–897, december 2001.

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

[40] P.M. Newman and J. Leonard. Pure range-only sub-sea SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1921–1926, Taipei, Taiwan, September 2003.

[41] P.M. Newman, J.J. Leonard, and R.J. Rikoski. Towards constant-time slam on an autonomous underwater vehicle using synthetic aperture sonar. In *Proceedings of the Eleventh International Symposium on Robotics Research*, Sienna, Italy, 2003.

[42] T. Pavlidis. *Algorithms for graphics and image processing.* Computer Science Press, 1982.

[43] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer-Verlag New York, Inc., 1990.

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

[45] I. Tena, Y. Petillot, D.M. Lane, and C. Salson. Feature extraction and data association for AUV concurrent mapping and localisation. In *Proceedings ot the IEEE International Conference on Robotics and Automation*, pages 2785–2790, Seoul, Korea, May 2001.

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

[47] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, To appear., 2004.

[48] G. Weiss and E. v. Puttkamer. A map based on laserscans without geometric interpretation. In *Proceedings of Intelligent Autonomous Systems 4 (IAS-4)*, pages 403–407, Karlsruhe, Germany, 1995.

[49] S. Williams and I. Mahon. Simultaneous localisation and mapping on the great barrier reef. In *Proceedings ot the IEEE International Conference on Robotics and Automation*, New Orleans, USA, April–May 2004.

[50] S.B. Williams, G. Dissanayake, and H. Durrant-Whyte. An efficient approach to the simultaneous localisation and mapping problem. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 406–411, Washington DC, May 2002.

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