CRANFIELD UNIVERSITY

MATHIEU ISSARTEL

TOWARDS A ROBUST SLAM FRAMEWORK FOR RESILIENT AUV NAVIGATION

CRANFIELD DEFENCE AND SECURITY

PhD
Academic Year: 2016 - 2019

Supervisors: Dr Lounis Chermak, Prof. Jean-Marc Le Caillec
Associate Supervisor: Prof. Mark Richardson
March 2021

CRANFIELD UNIVERSITY

CRANFIELD DEFENCE AND SECURITY

PhD

Academic Year 2016 - 2019

MATHIEU ISSARTEL

TOWARDS A ROBUST SLAM FRAMEWORK FOR RESILIENT AUV NAVIGATION

Supervisors: Dr Lounis Chermak, Prof. Jean-Marc Le Caillec
Associate Supervisor: Prof. Mark Richardson
March 2021

This thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Abstract

Autonomous Underwater Vehicles (AUVs) are playing an increasing part in modern navies, to the point that the control of oceans will soon be decided by their strategic use. In face of more complex missions occurring in potentially hostile environments, the resilience of such systems becomes critical. In this study, we investigate the following scenario: how does a lone AUV could recover from a temporary breakdown that has created a gap in its measurements, while remaining beneath the surface to avoid detection? It is assumed that the AUV is equipped with an active sonar and is operating in an uncharted area. The vehicle has to rely on itself by recovering its location using a Simultaneous Localization and Mapping (SLAM) algorithm. While SLAM is widely investigated and developed in the case of aerial and terrestrial robotics, the nature of the poorly structured underwater environment dramatically challenges its effectiveness. To address such a complex problem, the usual side scan sonar data association techniques are investigated under a global registration problem while applying robust graph SLAM modelling. In particular, ways to improve the global detection of features from sonar mosaic region patches that react well to the MICR similarity measure are discussed. The main contribution of this study is centered on a novel data processing framework that is able to generate different graph topologies using robust SLAM techniques. One of its advantages is to facilitate the testing of different modelling hypotheses to tackle the data gap following the temporary breakdown and make the most of the limited available information. Several research perspectives related to this framework are discussed. Notably, the possibility to further extend the proposed framework to heterogeneous datasets and the opportunity to accelerate the recovery process by inferring information about the breakdown using machine learning.

Acknowledgement

The research project carried out during a PhD is a unique experience. It ultimately confronts us to uncharted territories and continuously tests our resilience to adversity. Fortunately, unlike the AUV in our research topic, we are not alone in this journey.

I feel grateful for the support and advice of my (international) supervision team. I would like to warmly thank Dr Lounis Chermark for his dedication towards his students and his mentoring. I am also thankful for Professor Mark Richardson guidance and help. Working within the Cranfield University research facilities at the Defence Academy of the United Kingdom is a unique experience in a lifetime and I have really appreciated my time there.

I would also express my gratitude to Professeur Jean-Marc Le Caillec for his supervision while in France. Working with Jean-Marc has been the opportunity to deepen my appetite for probability and statistics. I would also like to thank him for supporting me during my first teaching and academic writing experiences.

This project would not have been possible without the funding provided by the Direction Générale de l'Armement as part of the PhD exchange program with the Defence and Science Technology Laboratory. I would like to thank Dominique Fattaccioli, Callum Meredith and Philippe Pouliguen.

I would also like to thank Professor Christian Person for his support and his implication in the IMT Atlantique PhD Student Society. I am also grateful to Professor Gilles Coppin for his involvement in the LabSTICC. Furthermore I keep a great memory of my visit at the research laboratories on underwater robotics of ENSTA Bretagne. I would like to thank Isabelle Quidu and her team for their time. I would also like to thank Naval Group for offering me the opportunity to further explore the challenges of the ocean.

This journey would not have been possible without the unfailing support of my family and my friends. My warmest thoughts go to my mother, Florence, my father, Eric, my sister, Carla and Caroline. I have a very special thought to Jacques and George Teste that gave me the strength to overcome some of the greatest challenges I faced. Of course, this experience would not have been as thrilling if my friends were not there: Petit Pierre, Grand Pierre, Alex, Guillaume, Louise, Diane, Jeremy, Carlos, Sonia, Oscar, Mathilde, Benjamin, Nicolas, Jean-Jacques, Nicolas, Lou as well as Marco and Eddie.

Table of Contents

List of Figures

List of Tables

Chapter 1

General Introduction

Autonomous naval systems have boomed considerably over the last decade. Whether they operate on the sea surface, in the air or in the depths of the oceans, they play an increasingly prominent role in modern fleets. They can provide a support role, such as in the fight against underwater mines, or be sent on scouting missions.

Increasing their level of autonomy is at the heart of today's research issues. It is based on a very diverse set of technical fields such as navigation, artificial intelligence, energy management, perception of the environment, and operational resilience. The missions undertaken by these systems are also becoming much more demanding and complex. The development of very long-range AUVs for strategic missions (anti-submarine warfare, area denial, nuclear deterrence, etc.) is currently shaping the future of naval engagement. This is accompanied by operational constraints such as constant stealth, which forbids any return to the surface in order to escape aerial detection.

The critical nature of such missions requires confidence in the ability of the autonomous system to deal with adverse situations that may impact some of its capabilities, without jeopardizing the mission. Operational resilience is a particularly complex issue in the case of underwater systems. More specifically: how to deal with an incident that could temporarily incapacitate some of the system's functionalities? Indeed, assuming that the number of long-range AUVs will thrive, systems for countering and intercepting these devices may constitute future threats that try to target and damage some of the AUV's components or instruments.

One of the most significant risks faced by autonomous vehicle is to get lost as a result of a failure. Such risk is exacerbated if the system is operating in an area not previously mapped and if the disruption introduces a "gap" in the processing of navigation data. If the device is lost, it can be captured or simply sink after running out of power. Based on this scenario, if one wishes to assign increasingly sensitive missions to such underwater systems, it is imperative to have confidence in their ability to manage unforeseen adverse situations and be able to securely return to base, for example.

This issue is at the heart of the research work presented in this thesis. The aim is to enable an autonomous underwater system to relocate itself within its environment after a temporary interruption of its acquisition systems caused by an operational failure.

The vehicle is assumed to have an inertial unit and an active sonar, a Side Scan Sonar, which allows it to observe the seabed from port to starboard. Using this data, it is possible to produce a representation of the environment which can also be used as a map for navigational purposes. This problem is widely discussed in the robotics literature as part of the SLAM (Simultaneous Localization and Mapping) algorithms category. While this technology has demonstrated significant results in traditional robotics, significant gaps remain in its application to the underwater environment. For instance, as the seabed is a loosely structured environment, dealing with a relocation problem is complex and can fail, especially if an operational issue occurs. Therefore, the resilience of such method is highly challenged if data association is complex and the INS drift becomes important.

This study is at the interface of three different area of expertise: **underwater navigation, lateral sonar imaging and robotics.**

In the first place, contemporary underwater navigation is based on various important assumptions that differ according to the operational context. For example, some systems rely on pre-existing maps and locate themselves relatively to these maps by seeking correlations between live observations and the *a priori* knowledge about the area. This is not always straightforward, as the marine environment is always changing due to various natural phenomena (currents, wildlife associated with low environmental variability and few key landmarks). The most important subsystem of the vehicle is its INS (Inertial Navigation System). The accuracy of the Inertial Measurement Unit (IMU) determines how long it is possible to navigate by dead reckoning without accumulating excessive uncertainty. Unfortunately, even a very accurate (and significantly expensive) unit will ultimately drift over the long run. To bound this uncertainty, a set of additional sensors or sources of information are used. It could be buoys or acoustic beacons, a GNSS signal or a map that is used in conjunction with a sensor such as a camera or sonar.

The objectives of the mission often define the type of tools that can be used for navigation. In the case of a stealthy operation, for example, it is impossible to go back to the surface to pick up a radio signal. The research problem of this thesis imposes the need for a self-contained navigation system, as the AUV cannot surface and must operate alone in uncharted territory. Consequently, only one type of navigation is applicable: the one addressed by the SLAM approach. Chapter 2 discusses the different types of underwater navigation and associated data fusion methods. As opposed to other use cases such as autonomous cars or general land-based robotics, the marine environment presents challenges that make the application of a SLAM algorithm particularly complex. Nevertheless, this is also the only solution given the operational constraints that defines our research topic.

The observation of the environment is at the centre of the solution of this technical problem. In particular, the family of active sonar sensors plays a key part. They are particularly useful for covering large areas due to the very good propagation of acoustic waves in water. In this study, we focus on side-looking sensors that scan from side to side of the vehicle, orthogonally to the heading. Lateral observation is not without difficulties for analyzing and representing the environment, which is crucial to be able to build a map for a SLAM based navigation.

First of all, it is necessary to perform the image formation using slant range measurements. This step requires to know the exact position of the carrier since it is necessary to project the acquisitions to form a mosaic. In the past, sonar images have been used to compensate for the drift of inertial units. This was then a problem of local registration, where the spatial relationship between two images that were assumed to partially overlap had to be determined. Iconic (underlying probability distribution driven analysis) and symbolic approaches (rather oriented in the semantic characterizations of objects salient in the environment) are the two usual families used in the context of local registration.

However, the operational context described previously is in fact a problem of **global registration**. The system failure makes the initialization of the vehicle position after the system resumes its activities extremely unreliable. Only a few parameters such as the failure duration and the post-recovery heading are immediately available. As a result, the search range for relocating the AUV becomes so large that traditional methods become less efficient and may introduce errors. In addition, the inboard calculation time dramatically increases.

Chapter 3 aims to address aspects of SSS sonar image analysis in the context of the global registration process. To do so, a two-step multi-scale approach is suggested. First, data associations are determined to produce an initial topology of the scene. Eventually, the aim is to get back to a local registration problem for finely registering the acquisitions later by limiting as much as possible the uncertainty on the position of the vehicle after the resumption of activities. In particular, attention is being paid to how the particularly exhaustive nature of certain iconic methods can be overcome by trying to limit their use to the study of certain types of seabed regions. Symbolic treatments are also adopted in order to use specific macroscopic structural information. It should be noted that image analysis does not fully solve the problem of relocation. Indeed, the production of false positives during this phase is inevitable due to the uncertainty of the sensor position. The consistency of the map resulting from a set of associations must be evaluated in order to validate the repositioning of the system. This data fusion step based on both inertial measurements and imagery is discussed in Chapter 4, through the use of a robust SLAM algorithm.

Creating a map on the basis of observations acquired during the mission requires the ability to locate these data. To do this, it is necessary to know the vehicle's trajectory over time. This estimate of the trajectory is itself dependent on the map because of the inertial unit's drift. Consequently, it is a joint estimation problem that must be addressed simultaneously, hence the SLAM modelling. One of the greatest vulnerabilities of the SLAM is the problem of forming reliable trajectory loop closures. This involves determining the likelihood of associating data separated by a significant time interval (basically: it consists in detecting points that have been previously visited by the AUV). These loop closures constraints (LCCs) play a crucial role in the estimation of the map because they shape its topology. If this topology is erroneous, then any local adjustment becomes futile.

Our operational scenario is precisely a loop closure problem. Indeed, the breakdown creates a " gap " in the trajectory which translates into a sudden increase of the uncertainty about the system location. It can be lifted by detecting one or several correct LCC(s). In highly structured environment, sensors can acquire a significant amount of information by successively observing the same point (e.g.: a wall with a LIDAR). Some multi-hypotheses SLAM algorithms, such as particle filters, can be used with great efficiency to tackle this problem. However, in the case of a side-looking sensor, not only a point is seen very infrequently (since an area is being swept on the sides of the vehicle), but also the aspect of a landmark varies significantly with the sonar grazing angle. Consequently, data association is rare and uncertain. As a result, it is critical to focus on determining the overall structure of the scene and to limit the uncertainty resulting from the failure to the maximum through different modeling. One of the great advantages of graph SLAM algorithms is their adaptability. By removing or adding a few edges it is easy to adjust the topology of the scene. Different approaches exist in the literature to take advantage of this optimization-based estimation process to make it more robust to outliers.

The purpose of Chapter 4 is to explore through our dataset how these algorithms behave in the very specific case of our scenario. A particular focus is devoted to the problem of undesired outlier dominance that can occur when the uncertainty resulting from the failure becomes large. This chapter discusses the use of robust SLAM algorithms using graphical modeling and describes their behavior in particular with respect to the disruption of the motion signal and the presence of erroneous LCCs. The conclusions of this part inform the design of our proposed framework, that is further developed in the next chapter. In particular, these examples highlight the need to perform a validation phase by a posteriori mosaic analysis to reject erroneous topologies.

Although the operational scenario is rather demanding, the results of the previous experiments suggest that the graph, that models the map and the trajectory, is made of blocks, or regions, of different nature. For instance, when the system is fully operational, the INS plays a predominant role. Most of the uncertainty is accumulated during rotation phases. However, in order to make an accurate measurement using a side scan sonar, it is necessary to keep a locally straight trajectory to avoid gaps in the mosaic. These phases constitute an example of consecutive blocks of constraints/vertices that have a locally low uncertainty.

On the other hand, the failure phase is an area of the graph topology where the uncertainty is locally very high. This "black-out" region can be treated via a black box approach where a set of parameters related to the failure is passed as input in order to narrow down as much as possible the region in which the system moves. Since the map estimate is produced by a least-squares optimization algorithm weighted by the level of uncertainty, it is analogous to a mass-spring system where some regions of the graph are more flexible than others. The reconnection of the trajectory that is made necessary by the failure is the most important “degree of freedom” to determine the correct topology of the scene. A SLAM framework that clearly segments the different regions of the graph in order to evaluate the effects of their respective parameterization is conceivably important.

This is also true for LCCs, whose consistency needs to be assessed a posteriori in order to validate the scene topology. Chapter 5 details the framework that has been developed in the course of this study to take advantage of this representation. A block diagram of the code that has been developped during our research is presented as well. The various implementations of the graph constraints are detailed as the graph needs to offer a certain modelling diversity in order to be transposable to different modalities of environmental observation. Furthermore, the framework is designed to be iterative as its objective is to eventually converge towards the ground truth. The part played by each of the main blocks of this framework is described. In particular, we present the tools that have been implemented to assess the map, which is represented by a multi-layers dataset, and detect potential inconsistencies that would be responsible for corrupting the graph topology.

The innovative aspect of this method is to no longer see this problem as a maximum a posteriori search that would mainly depend on LCCs, but rather to see the graph as a structure with clearly differentiated regions that are driven by hyperparameters that act on the topology of the scene.

Finally, the last section of this thesis presents the general conclusions of our research. In particular we discuss our findings on sonar data association under the global registration problem, the limits of traditional robust SLAM techniques and why our proposed Framework offer a novel methodology for addressing demanding and complex underwater missions. We also discuss the prospects for further development and improvement. For instance, some of our observation on the detection of keypoints could pave the way to further investigation to develop a sonar-specific keypoint detector for global registration problems. Moreover, the possibility of complementing our framework with more detailed work on the blackout zone is discussed, using a finer parameterization that would take into account considerations related to vehicle mechanics, some log data related to the failure or hydrodynamics. This would be in line with the previously described philosophy. The prospect of using Machine Learning algorithms that can map a complex correspondence between a small number of parameters and a possible range of trajectories for the blackout phase is also raised.

Chapter

2

The Navigation Problem

Highlights of the Chapter

- While diverse strategies have been developed in the past decades for enabling accurate underwater navigation, fully autonomous navigation remains a challenging task. The aim of this chapter is to explain what technologies are required for achieving this objective and demonstrate the need of SLAM based approaches.

- Due to the self-contain nature of Inertial Navigation Systems, an important part of the chapter is dedicated to explaining what instruments are used by AUVs to measure their kinematic properties in order to estimate their trajectory over time. More importantly, the INS sensors error model is discussed as their accuracy dramatically impact the estimation process for the AUV pose.

- This estimation problem is addressed by different algorithms that operate a fusion between the different sensors' measurements and a motion model. Kalman filtering being ubiquitous, we discuss the common implementations frequently found in underwater systems while also presenting other popular approaches like the Particle Filter.

- Unfortunately, as INS error is not bounded over time, additional sources of information are required for reducing the uncertainty about the AUV location. As a result, other navigation strategies, namely acoustic and geophysical approaches, are used for limiting the drift of the INS estimate. Our operational scenario, falls in the later category: as SLAM is using natural landmarks, the use of a sonar imager to probe the environment is required. A quick review of acoustic and geophysical navigation is presented.

- The last part of the chapter discusses the challenges we are addressing in this PhD with regards to the current state of the art for eventually achieving robust and resilient underwater autonomous navigation. Underwater SLAM is currently an intense field of study due to the unstructured nature of the seafloor. As fully autonomous navigation in uncharted environment cannot rely on a prior baseline, it is necessary to develop a robust SLAM navigation scheme.

2.1 Problem Statement

2.1.1 Operational Requirements

In order to fulfill its mission, the Autonomous Underwater Vehicle has to derive a reliable estimate of its location at any time. The accumulation of an excessive amount of uncertainty may result in dramatic consequences. First, the acquisitions may be georeferenced with a significant error, which would result in misaligned images. More dangerously, an autonomous payload may completely miss the mission recovery point and could be permanently lost. Like in any other navigation problems, corrections over the trajectory are performed, using all available tools. However, the underwater environment offers a challenging operational context. As the seafloor is highly unstructured, finding reliable anchor points is a complex problem. Furthermore, the lack of accurate modelling of the underwater perturbations like currents or wildlife, introduces additional uncertainty in the payload motion estimate. Last but not least, in some specific scenarios , the sensor may be unable to surface and cannot acquire a GNSS signal to precisely update its location.

Different navigational strategies are possible and they mostly depend on the operational context and the available inboard instruments. *Inertial navigation* relies on the use on the payload sensors that measure the kinematic variables of the vehicle. It is mostly a sensor fusion problem that consists in finding an appropriate *trade-off* between a *motion model* and a set of *measurements*. *Bayesian filtering* approaches like *Kalman filters* offer a solution to this problem. Unfortunately, inertial navigation inevitably accumulates too much error over the long run. In order to solve this issue, the underwater vehicle may use a baseline to correct its trajectory estimate. Most of the time, acoustic beacons serve as reference points. They can be mounted over a surface ship or attached to the seafloor. When this equipment is available, the trajectory correction is a straightforward problem. However, there are several main disadvantages to *Acoustic Navigation*. First, this is an onerous solution: deploying beacons may be expensive and their reach is limited to a few kilometers. But they also lack stealthiness. Surface ships could be very easily detected from afar and acoustic beacon signals may also be heard from a long distance as they produce omniderectional waves. Furthermore, any beacon relying on a GNSS signal (floating devices) to provide an absolute localization may be also under the threats of cyber attacks (GNSS spoofing). The advent of more constraining operational scenarios, coupled with the increasing capabilities of inboard equipment push the development of another approach: the *geophysical navigation*. This method consists in detecting landmarks that serve as reference points for refining the sensor location. In this case, the AUV may operate in a known environment with already referenced features, using a map of landmarks. Generally, the mission planning step consists in defining a path from a landmark to another one, until the final objective is reached. The difficulty here is to be able to detect and match these landmarks with their *a priori* signature.

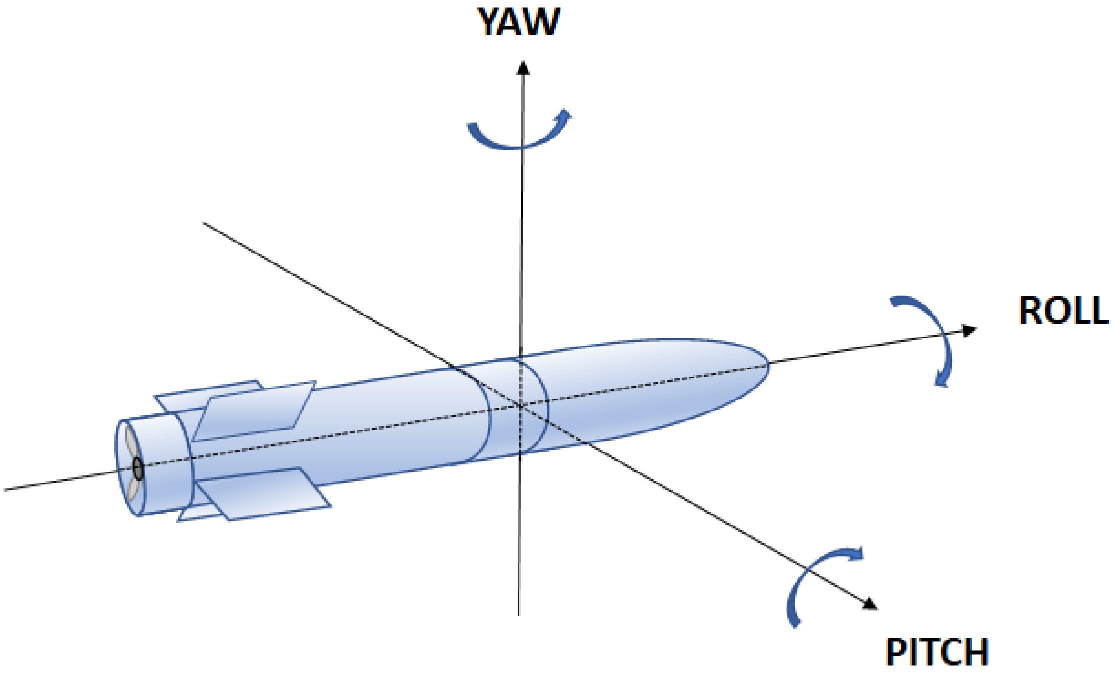


Figure 2.1: An AUV and its different axes of rotation.

The most challenging scenario is the case of autonomous navigation in uncharted seas. Here, the AUV has to build a map while navigating. This is by far the most complex case of study as the unstructured nature of the seafloor, coupled with the specificities of acoustic active sensors makes the detection and matching of unknown landmarks highly challenging. In order to address this scenario, it is critical to deploy robust SLAM algorithms coupled with sophisticated image processing techniques.

The aim of this part is to precisely define the navigation problem and review the different strategies available for correctly estimating the sensor trajectory. After presenting the key instruments that are used for motion measurement of the payload, the data fusion problem is discussed by presenting the Kalman Filter, the most famous algorithm derived from Bayesian filtering theory. More importantly, its limits are discussed with regards to the accuracy of inboard instruments. Ultimately, the operational scenario addressed in this thesis is carrefully discussed, with an emphasis over the robustness constraints.

2.1.2 Estimating the AUV Trajectory

The navigation problem consists in correctly assessing the vehicle location at all time in order to update its course according to the mission objectives. As the sensor is moving in a three dimensional environment, 6 parameters are needed to fully characterize the sensor *pose*: the three Cartesian coordinates and the three Euler angles (Roll, Pitch and Yaw as shown in Figure 2.1).

Estimating these parameters requires to define a *motion model* that states the relation between the kinematics variables and the set of mechanical constraints that apply to the system. For instance, in the case of a ballistic shot, the knowledge of the initial velocity with the mass of the projectile is enough to estimate the trajectory using *Newton's laws of motion.* Autonomous systems also perform *control actions* in order to adapt their trajectory across time. These actions are basically internal mechanical constraints and generally result in a change of velocity or attitude. Consequently, they are part of the motion model as they basically reflect the expected change between two consecutive poses. For instance, a pilot may change its plane pitch to gain altitude by updating the elevator state. As long as all the constraints to the system are known, either internal or external, it is possible to *predict* the pose of the vehicle a few moments later. Generally, internal sensors like *Inertial Navigation Systems* or *Odometers* in the case of ground robotics, provide a live measurement of the effect of a control input and are used for estimating the next robot pose. This is a more accurate estimate compared to pure control data where transitory states could not be taken into account.

Unfortunately, the use of the sole motion model with inertial measurements ultimately lead to an highly inaccurate estimate of the trajectory over time. The rate at which the uncertainty increases is directly related to the inertial sensor quality. Inexpensive INS may very quickly provide unreliable estimates compared to expensive cutting-edge sensors 2.1. This high price range strongly impacts the design of the AUV. The choice is dictated by the operational needs. If the scenario requires a lone autonomous agent operating during a long time, an highly accurate INS would be necessary. On the contrary, if a swarm of expandable AUVs is deployed for the mission, less sophisticated equipment would be included in the AUV design which would imply a critical need for correcting the quickly increasing uncertainty.

Furthermore, some phenomena that can be very difficult to model, like underwater currents, may cause greater errors over long missions. Analogously to the case of the ballistic shot, this would be similar to neglecting the effect of viscosity and the Coriolis force for a very long shot. The accuracy of the motion model and its critical parameters play an important part in the correctness of a pose estimate.

Depending of the scenario, several navigation strategies are possible. In each case, Inertial Navigation is critical for providing a first estimate. Traditionally, two additional approaches, with different variations, are commonly used. Acoustic Navigation, that mostly rely on different sorts of beacons, and Geophysical Navigation. The next section reviews these techniques and discuss the challenges they raise.

Table 2.1: INS: Performance and Cost [37].

GRADE	Civilian	Tactical	Aviation	Marine
Example of use	Pedometers	Unmanned Aerial Vehicle	Military Aircraft	Submarine
Position Drift	N/A	> 10 nmi/hr	~1 nmi/hr	< 1 nmi/day
Price	< \$100	5~20k\$	~100k\$	> 1M\$

Note: 1 nmi [Nautical Mile] = 1.852 km

2.2 Different types of navigation

2.2.1 Inertial Navigation

Principle

The simplest and oldest navigation approaches are based on *dead reckoning.* It basically consists in estimating the new location of the vehicle by computing the distance travelled during a unit of time, using the speed and the heading at the previous location. As the exact values of these variables may slightly drift over time, this approach become highly inaccurate after a few iterations. A significant improvement of dead-reckoning navigation had been made by using Inertial Navigation Systems.

The key idea behind Inertial Navigation is to rely on instruments that measure acceleration of the vehicle along its three axes 2.1. The resulting velocity can be computed by integrating these measurements over time. However, these measurements are only valid within the local frame bounded to the vehicle. In order to compute an estimate of the displacement of the vehicle within the reference frame, the rotational motion of the local frame has to be taken into account before proceeding to time integration.

The core component of an INS is an *Inertial Measurement Unit* that includes a set of accelerometers and gyroscopes. In general, three accelerometers measure the acceleration along the local frame axes while three gyroscopes assess the rotational motion of the vehicle axes. The INS is completed by a *Navigation Computer* that performs the estimation process by integrating the measurements after computing the gravitational acceleration.

Different INS designs and sensor technologies

Oldest designs of INS rely on *Stabilized Platform Techniques.* The inertial sensors are mounted over a platform that is kept aligned with the inertial reference frame by using a set of three gimbals, one for each axis of rotation. When a gyroscope detects a rotation, torque servos are activated and rotate the gimbals to keep the central platform in a constant orientation state with respect to the reference frame. As a result, the measurements made by accelerometers are always done along the same directions across time. This type of INS design has the advantage of

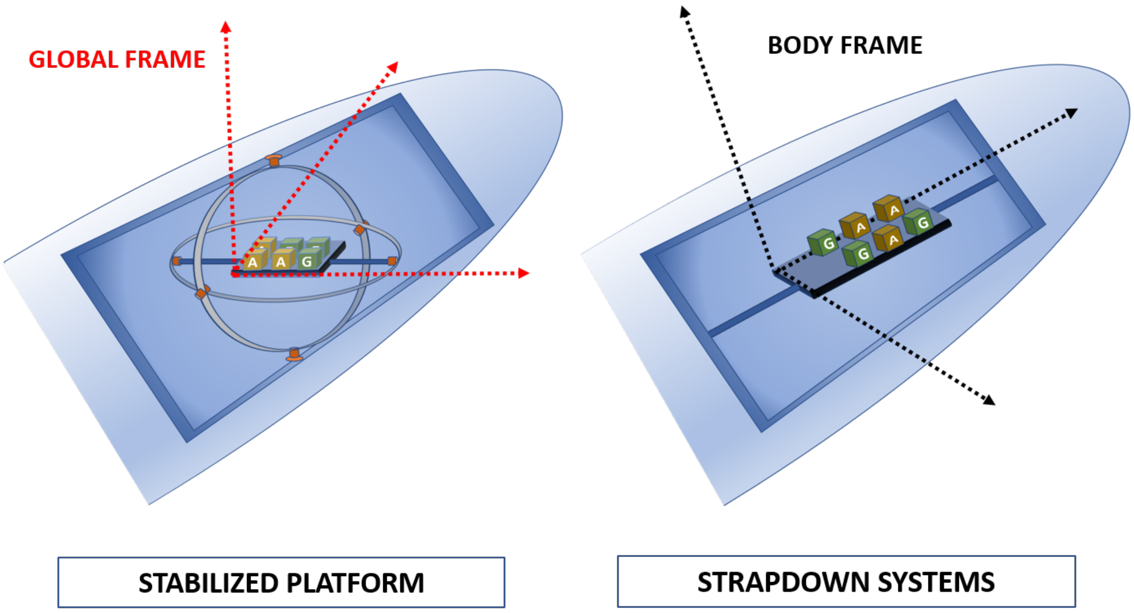


Figure 2.2: Different types of INS designs (A: Accelerometers, G: Gyroscopes).

providing accurate and computationally simple estimates of the vehicle location. Unfortunately, these large and heavy sphere-shaped structures are mechanically complexes and expensive. Furthermore, due to numerous mobile parts and the risk of *Gimbal Lock*, they are fragile and may fail during operations.

Nowadays, modern INS use a radically different approach called *Strapdown INS*. Instead of mechanically isolating the inertial sensors, they are now directly attached to the vehicle structure. While these INS are smaller, cheaper and mechanically simpler than the *Gimbals INS*, they do not directly measure the acceleration in the reference frame. As a result, it is necessary to perform a frame transform after deriving the vehicle attitude vector. This enables to find the three acceleration components over the inertial reference frame and run the Inertial Navigation equations. Not only this additional processing step increases the computational complexity, but it accentuates the measurement errors as they propagate through several operations. A small tilt error over the estimate of the attitude vector results in an erroneous projection of the acceleration vectors, that are integrated twice over time. Furthermore, the gravitational corrections of the navigation computer are also incorrectly applied. Both INS designs are presented in Figure 2.2.

This is the reason why the main source of error in INS navigation comes from the gyroscopes. Consequently, it is critical to deploy highly accurate and robust sensors in order to avoid a significant drift over time. In the past, gyroscopes were mechanical tools using a system of gimbals coupled with a spinning wheel. Because of the conservation of angular momentum, the wheel maintains its initial orientation inducing a rotation of the gimbals . This rotation can be measured by reading the angle pick-off between gimbals. A straightforward calculation provide an estimate of the rotation of the whole system. As well as being fragile and prone to locking, these sensors may drift over time due to friction. They have been conveniently replaced by optic based solution relying on the *Sagnac Effect*. There are two categories of sensors: *Fibre Optic Gyroscopes* (FOG) and *Ring Laser Gyroscopes* (RLG). Both are basically one-path Sagnac interferometers where light pulses follow a path over a plane, either following an optic fibre coil or a mirror path. Two coherent light pulses are emitted in opposite directions and interfere at the end of the system. When the system is rotating in the interferometer's plan, a short difference of optical path occurs which results in a phase shift between the two pulses. The analysis of the interference pattern at the end of the system allows to measure the phase shift which is related to the rotational rate of the motion. Indeed, compared to mechanical gyroscopes that measure angles, optical systems measure an angular speed, which means that the signal and its noise are integrated. Fortunately, these sensors largely outperform their mechanical counterparts and are a lot more robust against mechanical stress as there is no moving parts. Unfortunately they are very expensive. A third category of instruments relies on the use on Microelectromechanical Systems (MEMS) technologies. Here, small vibrating components are used to measure the Coriolis effect that result from a rotation of the system. While being a lot less accurate than optic based technologies, MEMS sensors offer interesting properties. On top of being cheap to manufacture, they are small and light. Furthermore, they also are robust against mechanical constraints and highly reliable, making them interesting tools for designing low-cost autonomous platforms. Right now, they do not offer the accuracy required for navigation-grade requirements, but they may be used in conjunction of additional tools for correcting their uncertainty.

INS sensors error model

Estimating the accuracy of such instruments is a complex problem as several sources of error simultaneously apply to the system. In general, three different types of errors are considered for modelling sensor uncertainty. The first category is an offset that is applied to the true signal value called a *bias*. This is an additive error that grows either linearly with time for gyroscopes (one integration) or quadratically for accelerometers (two integrations). The effect of the instrument bias can be measured with calibration tests. Unfortunately, the bias is not an instrument constant. It varies between two operations and during the mission. In order to measure the variations of the bias, two parameters are required. *Bias Repeatability* is a measure of the variation of the bias offset from a mission to another one. The lower variability, the better.

Furthermore, during long mission, some physical parameters like the temperature may change. Consequently, the properties of the instruments may be impacted. For instance, the length of an optic fiber may slightly change across time, leading to a variation of the bias. This type of bias change is called *Bias Stability*. This parameter measures how the bias is expected to change during an acquisition. As an offset, biases are independent from the measured signal. The second category of error lays in the relation between the input signal value and the output produced by the sensor. In fact, the measured value is not exactly the value of the force that was applied to the sensor.

Table 2.2: Example of Modern High-Grade INS Accuracy (source: iXblue)

REFERENCE	Heading Error	Error after 60s	Error after 120s
Rovins Nano (Entry Level)	0.05 deg RMS	0.6 m	2.2 m
Phins Subsea (Cutting Edge)	0.01 deg RMS	0.06 m	0.3 m

The *Scale Factor* measures the linear relation between these two signals. The linearity hypothesis may also be erroneous due to non-linear phenomena (due to saturation for instance). In that case, more complex corrections may be applied. On top of these previous categories, a random white noise contaminates the measured signal. In the literature, this noise is refereed as a *Random Walk* as the noise integration lead to Random Walk stochastic process. Other sensor-specific sources of noise also exist. If electronic sensors are used, an additional *Flicker Noise* impacts the Bias Stability over time (a common issue while using MEMS instruments). Misaligned sensors may also produces correlated measures as the measurements are not exactly performed over the local frame axis. This last issue could be mitigated by improving the manufacturing standards or by performing calibration corrections.

Experimentally, it is possible to estimate the effect of these different sources of error by performing an *Allan Variance* analysis [15]. Given a long measurement, the Allan Variance test consists in splitting the sequence into a subset of consecutive bins of duration τ. Then, the signal f is averaged for each bin and the following estimator is computed:

$$\sigma^2(\tau) = \frac{1}{2\cdot(n-1)} \sum [avg(f)(\tau)_{k+1} - avg(f)(\tau)_k]^2 \tag{2.1}$$

After testing different magnitude of τ, it is possible to plot a log-log evolution of $\sigma(\tau)$ as presented in Figure 2.3. This plot exhibits different regions where a specific source of error prevailed for a given characteristic duration. This analysis is critical for separating the different sources of noise that simultaneously contribute to the overall system error. Two interesting notes: first, the shorter the duration the more bins are usable, which means that the test is more statistically significant for small characteristic times. Conversely, in order to measure the effects of a random process with a large characteristic duration, it may be necessary to proceed to a long experiment in order to produce enough bins for a meaningful test. Second, the minimum of the Allan Variance analysis is used as an estimate of the *Bias Instability* parameter. This is the source of one of the INS performance indicator that is advertised by the manufacturers. An example of an Allan Variance study of MEMS technologies, with a short introduction to INS Navigation Systems can be found in [106].

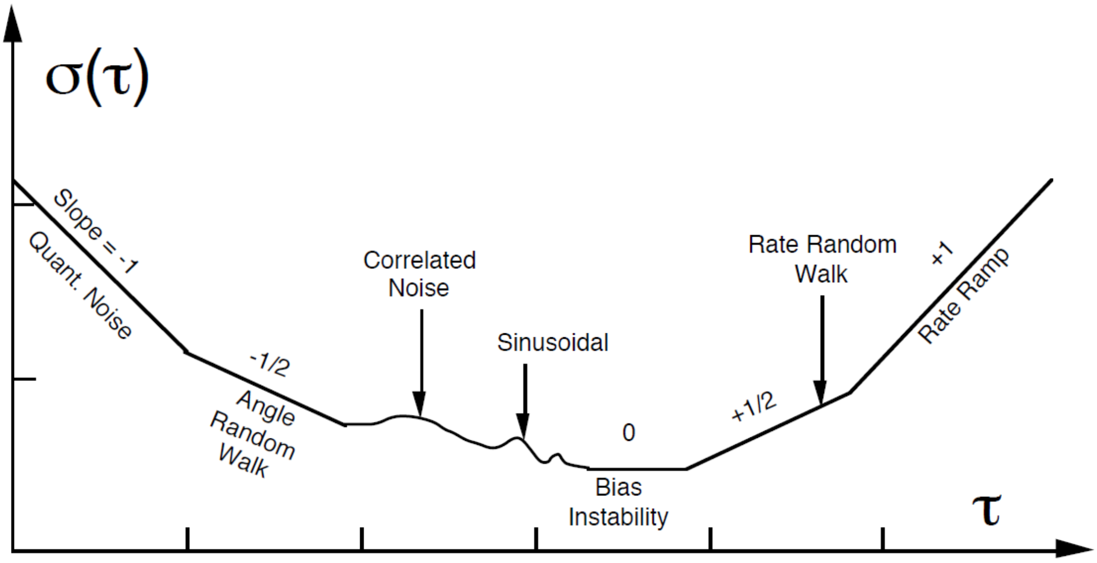


Figure 2.3: Classic Log-Log plot of Allan Variance analysis. τ = Sampling Duration , $\sigma(\tau)$ = Deviation for a given Sampling Duration. [15]

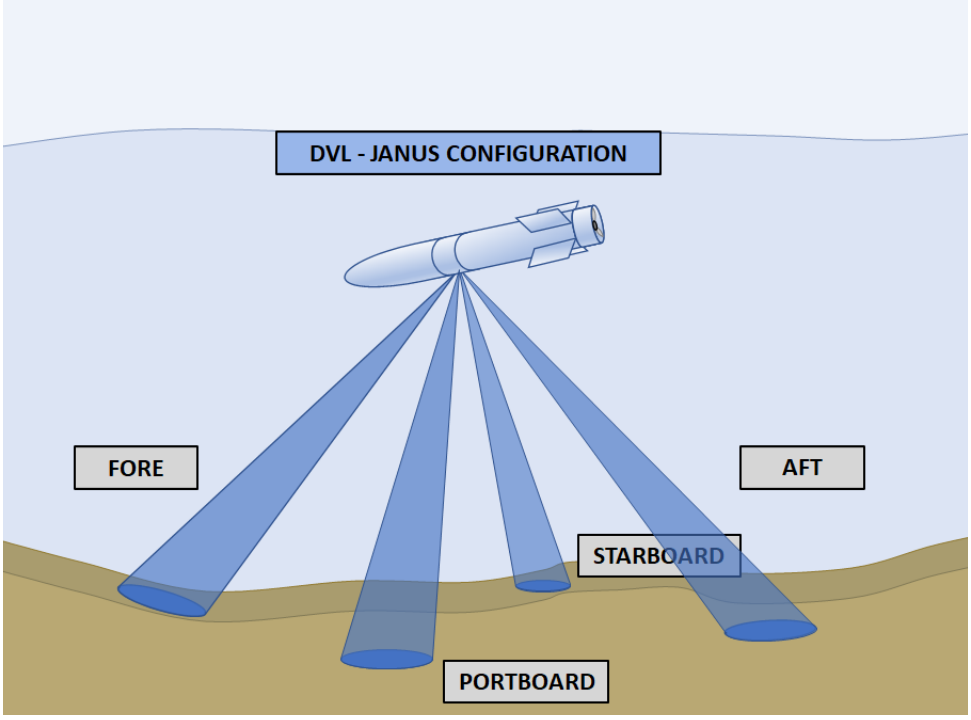


Figure 2.4: DVL in a Janus Configuration.

Completing the INS instruments

In general, it is quite rare to only rely on an INS. As previously stated, the price of high performing inertial sensors may be too expensive for the AUV design. As a result, Inertial Navigation has to be assisted by additional sensors in order to prevent a drifting and biased estimate of the vehicle velocity and heading.

Whereas an INS has to perform a series of integration in order to compute the AUV's speed, a *Doppler Velocity Log* (DVL), provides a direct measurement of the speed. A DVL is an acoustic sensor that uses the *Doppler Effect* for estimating the payload's speed relatively to the ground. Like in the case of radar technologies, it consists in measuring a frequency shift between two signals. A transmitter located below the payload emits an acoustic beam towards the sea bottom at a given frequency. Once the beam hits the ground, it is reflected towards the payload. The backscattered wave is measured by the DVL receiver. It is important to note that the AUV is successively a mobile transmitter and receiver. If the sensor is moving in a direction parallel to the wave propagation axis, the frequency shift will be given by [16]:

$$\Delta f = \pm \frac{2 v_{AUV} f_{DVL}}{c_{Sound}} \tag{2.2}$$

The sign of the shift depends if the AUV is moving in the same direction as the emitted wave (positive) or is the opposite direction (negative). In practice, a set of four transducers displayed in a *Janus Configuration* (Figure 2.4) is required for solving a system of equations that gives the three velocity components in the AUV's body frame. Like in the case of INS instruments, similar sources of errors occur (white noise or sensors misalignment for instance). While calibration tests are generally performed in order to derive correction terms, the speed estimate still carries uncertainty resulting from complex operational factors. One operational limitation of the use of a DVL is to keep locked to the sea bottom. As this requirement is not a problem when performing shallow surveys, it becomes a critical issue in deep waters. The operational range of a DVL mostly depends of the frequency of the emitted signal. High frequencies ($\sim$ 500 KHz) have a typical range of a couple hundred meters while low-frequency DVL ($<$ 100 KHz) can track the sea bottom up to 1000 meters depths [52]. Unfortunately, DVLs with a long range require a significant amount of power. This could be a problem for some AUV designs. In deep operational environments, it may be possible that the seafloor is not in range during some phases of the mission. In that case, another type of instrument, working on the same principle as the DVL can be used. *Acoustic Doppler Current Profilers* or ADCPs, perform a measurement of the water currents relatively to the vehicle frame. Thanks to this instrument, it is possible to measure the AUV velocity while the vehicle is too far from the sea bottom. Water tracking using an ADCP is very useful for monitoring the descent phase preceding a survey in deep waters [93].

Furthermore, INS can be assisted by different types of compasses, in order to provide an additional estimate of the AUV heading [33]. It is either possible to rely on *magnetometers* which consists in using the Earth's magnetic field in order to keep track of the magnetic North. As geographic North and geomagnetic North do not coincide, it is necessary to take into consideration the *magnetic declination* in order to compute the heading within a frame bounded to the geographic North. One downside of magnetic based instruments is that the local magnetic field is locally disturbed by the presence of ferromagnetic material, including the AUV itself. In order to avoid such problem, it is possible to use a gyrocompass that relies on the Earth's Rotation. Due to *gyroscopic precession*, it is possible to keep track of the Earth geographic North direction. Consequently, an estimate of the vehicle heading can be derived.

So far, the different instruments we have presented were measuring specific kinematic variables of the AUV. However, it is important to note that in the literature, Inertial Navigation may also involve the use of a GNSS receiver when the AUV surface. This measurement performs an absolute localization of the payload which is critical for relocating the AUV after a long underwater phase. An interesting property of a GNSS measurement is that the uncertainty is constant compared to IMUs measurements that drift over time. Unfortunately, depending of the operational scenario, this source of information may not be available. Indeed, if stealth is required, surfacing may be prohibited during the operation in order to avoid to be detected. Even in the case of civilian applications, the presence of ice sheets may also prevent the AUV to surface as well. Furthermore, a GNSS signal can be jammed in denied regions which could also introduce a severe operational risk. If too much confidence is given to the GNSS signal, the payload can get lost and the mission compromised.

Summary of Inertial Navigation

Due to the self-contained nature of INS and especially IMUs, Inertial Navigation is the core component of underwater navigation. Modern IMUs sensors are robust, stealthy and immune to jamming. Unfortunately, they are prone to drift which result in too much uncertainty. Cutting-edge IMUs may provide a more stable estimate of the payload location during short mission but they may be either too expensive or too large for the AUV design. As a result, additional sensors like DVL are commonly used for providing additional measurements of the AUV kinematic properties. Consequently a critical estimation problem arise. Indeed, as several signals are generated by different sensors with various degrees of uncertainty and different sampling frequencies, the fusion process for generating a good estimate of the vehicle location is a sensitive problem. *Kalman Filtering* is the common process for performing such estimation. The next section discuss the underlying concepts behind the Kalman Filter, its common derivatives and presents several underwater applications of Inertial Navigation using this filter.

2.2.2 Sensor Fusion and State Estimation

Kalman Filtering

As stated in Section 2.1.2, the key navigation objective is to know the AUV *pose* (coordinates and attitude) at any time. Unfortunately, these variables cannot be directly observed. Instead, motion sensors measurements, in conjunction with the mechanical equations that apply to the system, are used for inferring the pose parameters. This raises a complex estimation problem where a hidden signal, the *system state vector*, has to be estimated using observed signals that include different sources of noise. The state vector encompasses all the key variables that are necessary for describing the state of a *dynamic system.* In addition to the kinematic variables (pose, velocity and acceleration), it may also include variables related to motion sensors measurement like biases for instance. The estimation process consists in deriving the best estimate possible $\hat{x}_t$ of state x_t using the knowledge of $\hat{x}_{t-1}$ and the set of constraints that apply to the system between $(t-1)$ and t. When the information carried by the state vector x_{t-1} is sufficient for producing the *optimal estimate* of x_t, the system is said to be *complete.* As a result, no other measurement or past data about previous states would carry useful information that would result in a better estimate of x_t. *State Completeness* is generally a theoretical requirement for most estimation algorithms, but in practice, only a few state variables are used which result in an approximation of the problem.

Estimating a signal from a noisy measurement had been actively addressed since the 1950s notably by Wiener [104] which laid the foundation for the Kalman Filter that was suggested in 1960 [49]. In order to define an efficient estimator of the hidden signal, it is necessary to define a *Loss Function* that measures the difference between the estimate and the true signal. The approach suggested by Wiener consists in deriving the *Optimal Estimator* that minimizes the *Mean Squared Error* as the Loss Function. He shows that this problem could be solved by designing an optimal filter with a specific finite impulse response. The complexity of deriving the filter parameters lead Kalman to present a different approach based on *State Transition* for computing the MSE estimator. Due to its operational efficiency, the Kalman filter had been a significant contribution to the system control community. Nowadays, several modern derivatives of the Kalman Filters are commonly used in the field of autonomous systems and serve as the key estimation and fusion algorithm for inertial navigation.

The Kalman Filter is a recursive *Bayesian estimator* that performs two specific steps for each iteration. The first phase consists in *predicting* state x_t using the previous state estimate $\hat{x}_{t-1}$ and the properties of the *State Model.* This model reflects the effects of the dynamical constraints that apply to the system between $(t-1)$ and t. It is represented by three matrices: the *State Transition Matrix* Φ_t, the *Control Command Model* A_t and the *System Noise Matrix* Q_t. The first one represents the *deterministic function* that describes the evolution of x_{t-1} as a dynamic system. The second matrix models the effects of the system control commands u_t. These commands represent how the AUV modifies its internal parameters to change its course.

Kalman Filter: The State Transition Model

The System State Transition between $(t-1)$ and t is described by the following model:

$$x_t = \Phi_t x_{t-1} + A_t u_t + q_t \tag{2.3}$$

- **State Vector**: x_t
- **State Transition Matrix**: Φ_t
- **Control Command**: u_t
- **Control Command Model Matrix**: A_t
- **System Noise**: q_t

Note: The Process Noise is assumed to be a **white noise source** and is described by the **System Noise Covariance Matrix**: $Q_t \sim \mathbf{cov}(q_t)$. Furthermore, the original Kalman algorithm model assumes a **linear** relation between x_t and x_{t-1}.

Kalman Filter: The Prediction Phase

The first phase of the estimation process consists in predicting the system state using previous estimates and the transition model:

Derivation of the **Prior State Estimate**:

$$\hat{x}_t^{'} = \Phi_t \hat{x}_{t-1} + A_t u_t \tag{2.4}$$

Derivation of the **Prior Error Covariance Matrix**:

$$P_t^{'} = \Phi_{t-1} P_t \Phi_t^T + Q_t \tag{2.5}$$

Note: The Error Covariance Matrix is defined as follow:

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

Actually, the INS performs a live measurement of the effects of such commands. As a result, it is more accurate to rely on the INS measurement in order to avoid the inaccuracy due to transitory phases (speed increments are not instantaneous for instance). This is why most of the time, u_t is simply the motion measurements provided by the INS. The last matrix, Q_t encompasses the uncertainty related to the system modelling. It may either model the noise of the different instruments used in this step or the part played by variables that cannot be measured. For instance, if acceleration cannot be measured, it may be modelled as a source of noise for the speed estimate.

The specificity of the Kalman filter is its ability to also maintain an estimate about the uncertainty over the state estimate. The *Error Covariance Matrix* P_t measures the accuracy of $\hat{x}_t$ and the correlation between the errors in the different state estimate variables. By iteratively updating P_t, it is possible to keep track of the accumulated error in the Kalman Filtering process. The outputs of the prediction step are two intermediary estimates: $\hat{x}_t^{'}$ and $P_t^{'}$. The *prime symbol* reflects the fact that they are not the final estimates of the next state as they do not contain information from the observation data.

The second step in the Kalman Filter consists in refining the intermediary estimates by using observation data. In the case of underwater navigation, this is the part of the sensor fusion process where data acquired by instruments such as the DVL or the GNSS receiver are used. These instruments generate a *measurement vector* z_t that provide an additional source of information about the system state. Each measurement can be expressed as a function of the system state altered by some noise. This relation is represented by two matrices: the *Measurement Model Matrix* H_t and the *Measurement Noise Matrix* R_t. Analogously to the prediction phase and the system model, H_t represents the deterministic relation between the observations z_t and the system state x_t (not the estimator $\hat{x}_t$), while Q_t models the noise related to the observation instruments. The measurement vector, using the measurement model, provides a different state estimate that results from observation sensors only. The gap between this estimate and $\hat{x}_t^{'}$ is called the *Measurement Innovation*.

Kalman Filter: The Measurement Model

The relation between a measurement vector and the system state is described by the following model:

$$z_t = H_t x_t + r_t \tag{2.6}$$

- **Measurement Vector**: z_t
- **Measurement Model Matrix**: H_t
- **Measurement Noise**: r_t

Note: The Measurement Noise is assumed to be a **white noise source** and is described by the **Measurement Noise Covariance Matrix**: $R_t \sim \mathbf{cov}(r_t)$. Furthermore, the original Kalman algorithm model assumes a **linear** relation between x_t and z_t.

The goal of the Kalman Filter is to compute a weighted average between these two estimates that favors the sources of information of higher certainty. Consequently, the parameter that tunes the weighting, the *Kalman Gain Matrix* K_t is derived from $P_t^{'}$, Q_t and R_t. In fact, K_t is the matrix in the KF derivation that guarantees that the filter produces the optimal estimator (lowest state uncertainty P_t possible) according to the Mean Squared Error function. In addition to deriving the next state estimate $\hat{x}_t$, the

Kalman Gain is also used for updating the Error Covariance Matrix by computing P_t.

A critical property of the Kalman Filter is that the state uncertainty quickly decreases over time thanks to the measurement data. Initially, the Kalman Gain is high, which means that more weight is given to the measurements as they are more reliable than the current state estimate. However, as the error covariance matrix is updated after each iteration, less and less importance is given to the measurement data. To some extent, the system is acquiring reliable information over time which results in a significant decrease in the uncertainty related to the state estimate. After some time, the Gain converges towards an *equilibrium* while P_t reaches a minimum. This is an interesting property for ensuring sensor fusion robustness. Indeed, the post update phase estimate x_t is less impacted by measurement noise over time which result in a more stable state estimate. Furthermore, if a sensor produces an *outlier*, its impact over the estimation process is neglected. Consequently, the *Convergence Rate* is a critical operational property of the Kalman Filter. This parameter is impacted by the experimental tuning of the different matrices that model the sources of noise, Q_k and R_k, and the initial state uncertainty $P_{t=0}$. It is important to emphasize that the Kalman Filter model assumes that the different sources of noise could be modelled as *White Noise*. In reality, this is an approximation. As discussed in the inertial instrument review section, some sources of noise are not memory-less and cannot be represented as an uncorrelated process. In practice, the choice of Q_k and R_k parameters should be significant enough to reflect this approximation, otherwise the state error may be largely underestimated.

Kalman Filter: The Measurement Update Phase

During the update phase, the Kalman filter first derive the Gain using the different variables that model the system uncertainty. In a second time, the posterior estimates are computed using the measurement model and the kalman gain.

Derivation of the **Kalman Gain**:

$$K_t = P_t' H_t^T (H_t P_t' H_t^T + R_t)^{-1} \tag{2.7}$$

Derivation of the **Posterior State Estimate**:

$$\hat{x}_t = \hat{x}_t' + K_t(z_t - H_t \hat{x}_t') \tag{2.8}$$

Derivation of the **Posterior Error Covariance Matrix**:

$$P_t = (I - K_t H_t) P_t' \tag{2.9}$$

Common Kalman Filter Derivatives

Theoretically, the Kalman filter produces the *Optimal State Estimator*, as long as a few conditions are met. First, it is assumed that the state transition and measurement equations are linear. In practice, if the trajectory cannot be described by linear equations like in the case of a circular or parabolic one, this condition is not met. Second, the system and observation noises are assumed to be generated by white noise sources. As previously discussed in the inertial sensor part, the motion sensor noise sources are complex and cannot be modelled as uncorrelated sources of noise. This is not a memory-free process and it may result in a critical drift of the estimation process over the long run. As a result, the original implementation of the Kalman filter could easily become inaccurate and sub-optimal.

The *Extended Kalman Filter* addresses the non-linear equation problem by performing a linearization of the measurement and state transition model using a first order *Taylor Expansion*. This requires to compute the *Jacobian Matrix* of the state transition and measurement non-linear functions, using the current state estimate. The EKF is perhaps the most widely used implementation of the Kalman Filter in the field of robotics. However, this filter does not produce an *optimal estimator* as the posterior state estimator results from an approximation. In practice, if the system is highly non-linear (if the linearization is not an acceptable approximation), the EKF could produce an inconsistent estimator and other approaches may be more suitable.

It is interesting to mention that in the field of probabilistic robotics [97], the objective is to derive the *Belief Function* of the Autonomous System. This probability distribution represents the system knowledge about itself and its environment. It basically encompasses the system pose and estimators related to the environment landmarks in SLAM algorithms. If the initial belief is assumed to be Gaussian, the linearity of the KF filter produces a *Gaussian posterior belief*. When transition and measurement function are not linear, the posterior belief cannot be described as a Gaussian belief. However, the EKF linearization of the equations consists in producing a Gaussian approximation of the posterior belief.

The Extended Kalman Filter

EKF System State Transition And Measurement Models:

$$\begin{cases} x_t = f(x_{t-1}, u_t) + q_t & (2.10) \\ z_t = h(x_t) + r_t & (2.11) \end{cases}$$

With f, h, non-linear and differentiable functions. The following matrices F, H, are their **Jacobians**, with regards to the state variable.

$$F_t = \frac{\partial f(x,u)}{\partial x}\bigg|_{x,u=\hat{x}_{t-1},u_t} \quad (2.12) \qquad H_t = \frac{\partial h(x)}{\partial x}\bigg|_{x=\hat{x}_t'} \quad (2.13)$$

EKF Prediction and Measurement Update Phases:

The EKF equations are similar to the ones presented in the Kalman Filter section. The differences are highlighted below:

$$\hat{x}_t' = f(\hat{x}_{t-1}, u_t) \tag{2.14}$$

$$P_t' = F_t P_{t-1} F_t^T + Q_{t-1} \tag{2.15}$$

$$\hat{x}_t = \hat{x}_t' + K_t(z_t - h(\hat{x}_t')) \tag{2.16}$$

Rather than performing an analytical linearization using a Taylor expansion that requires to compute the models derivatives, the *Unscented Kalman Filter* [46] relies on a statistical approach. *Sigma Points* are extracted from the prior state distribution and passed through the system's non-linear functions. These points are deterministically sampled around the distribution mean and are associated with a set of weights. The parameters of the posterior state distribution are computed using the weights and the images of the sigma points. Apart from the sampling process, the UKF prediction and update equations work in a similar fashion to the EKF ones.

Experimentally, the UKF is slightly more time-consuming than the EKF. On the other hand, the native implementation of the UKF can capture higher order terms of the Taylor expansion compared to the EKF (at least the second order). Both approaches are extensively discussed and compared in [101]. However, if the underlying state distribution dramatically diverges from a Gaussian, the UKF can poorly perfom. In that case, a better strategy consists in using non-parametric algorithms like the *Particle Filter*.

More generally, these KF derivatives assume that the AUV pose estimate may be described as a unimodal distribution. Unfortunately some sensors deliver measurements that cannot unequivocally characterize the payload location. For example, this issue arises with acoustic sensors that only perform a range measurement. Consequently, several likely locations for the AUV may exist at the same time until the spatial ambiguity is removed using additional measurements. In order to solve this problem in a Kalman Filtering fashion, *multi-hypothesis* techniques have been developed. First, the measurements are modelled as *Gaussian Mixtures*. Each component of the mixture represents a hypothesis and is associated with a *probability score*. A *Null Hypothesis* is also introduced in order to be able to reject very unlikely measurements.

They are different strategies in order to address a multi-hypothesis problem. They can be split into two categories. The first category includes *best-fix* and *weighted-fix* algorithms. They propagate a single state estimate between each filtering cycle. In the case of *best-fix*, a dedicated algorithm selects the hypothesis with the highest score and propagates it during the update phase in a standard KF. In the case of *weighted-fix* methods, the state estimate is computed using all hypotheses but weighting their contribution in the Kalman Gain computation using their score. The *Probabilistic Data Association Filter* [28] is an implementation of this method and is useful in order to address the landmark association problem in environments with a high rate of false alerts. Algorithms that compute several state estimates simultaneously fall into the second category. For instance, the *Multi-Hypothesis Kalman Filter* use several KFs in parallel, applying the different observation hypotheses to the several state estimates. After many iterations, erroneous measurement hypotheses are detected and eliminated. In order to avoid an increase of the number of possible state estimates between cycles, some hypotheses are either discarded or merged depending of their probability score. If it is necessary to derive a unique state estimate at a given time, it is possible to either select the more likely hypothesis or compute a weighted average using all the state estimates. The need for such type of algorithm originates from multi-targets tracking problems where numerous data associations hypotheses need to be assessed [84].

The Particle Filter

The Kalman Filter and its derivatives are part of the group of parametric fusion algorithms. They recursively derive a small number of parameters, namely the mean and the covariance, in order to model the system state variable probability distribution after each cycle. In some circumstances, this approach is insufficient for capturing more complex state distributions. The case of multi-modal distributions is more challenging but can be still addressed using techniques like the *Iterative Gaussian Mixture Approximation of the Posterior* [92]. This type of algorithm offers an integrated method for running a Kalman Filter and producing an estimate under a model of Gaussian Mixture. Unfortunately, some situations may require to run more versatile algorithms that do not assume that the posterior distribution could be approximated by a given model. Consequently, non-parametric estimation approaches, like the *Particle Filter*, became very popular [38], [34].

Here, the posterior distribution is approximated by recursively computing a set X_t of possible state vectors. Each vector $x_t^{[n]}$ of X_t is called a *Particle*. The Particle Filter is articulated around three steps. The first one consists in sampling a state transition distribution, using the control vector u_t. These samples represent the new possible states after performing the prediction phase. While the KFs incorporate the measurement update as a corrective term weighted by a gain, the PF computes an *Importance Factor* for each newly created particle. The heavier the weight, the higher the probability to observe the measurement given the state defined by the particle. The last step consists in sampling with replacement the current particle pool, using the weights as the probability to draw each particle. The ones with low weights disappear from the distribution while the others have many duplicates in the final pool.

The way the PF represents the posterior makes it very efficient at addressing multi-modal and complex distributions. However, it has a few practical downsides. First, it is necessary to maintain a high number of particles (several thousands) to guarantee that the posterior distribution approximation is satisfactory. In terms of inboard resource management, this has a significant cost.

Fortunately, most modern implementations of the PF have an adaptive management of the amount of particles. If the state uncertainty is low, it is not necessary to keep many particles as they are concentrated in the same region. However, if the vehicle face a serious localization problem, more options must be considered. A strong requirement for guaranteeing the effectiveness of the PF is to make sure that at least a few particles lay in the vicinity of the true state. Indeed, like any Bayesian inference problem, it is dangerous to consider a too narrow prior belief otherwise the algorithm could end up in a particle deprivation problem. In that case, a dramatically high particle population has to be generated in order to address the location uncertainty.

Last but not least, the random nature of the resampling phase also artificially introduces uncertainty if the AUV remains static. Indeed, the particle pool variance increases while no change should be observed. Fortunately, several resampling strategies exist to address this issue.

The Particle Filter - Part 1

Prior State Distribution:

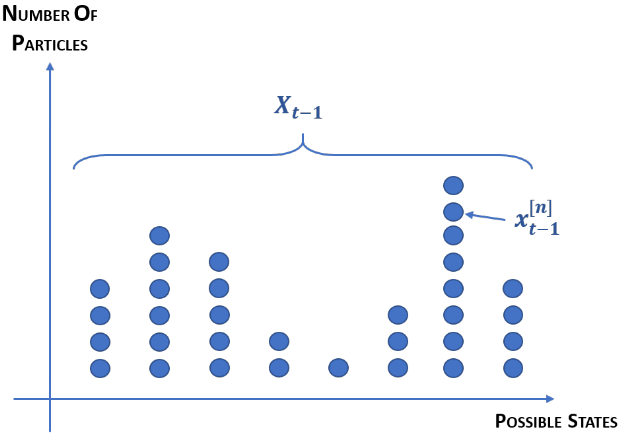


Figure 2.5: Particle Distribution Before A New Estimation Cycle

The *Particle Filter* approximates the state distribution using a vector X_{t-1} of N possible states called *particles*. Some of them are redundant as they represent more likely states.

$$X_{t-1} = [x_{t-1}^{[n]}]_{n\in[\![1\,..\,N]\!]}$$

State Transition:

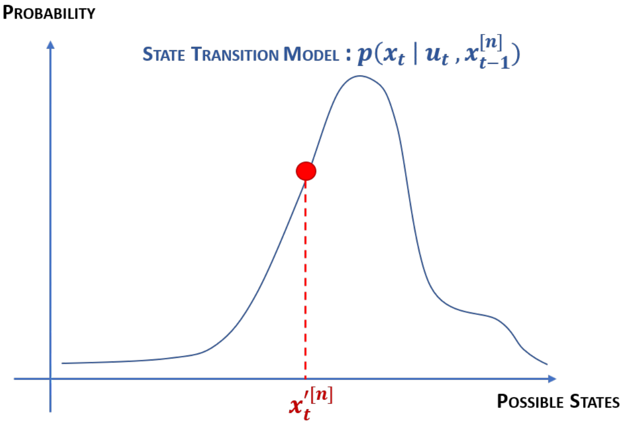


Figure 2.6: Sampling the State Transition Distribution

For each particle $x_{t-1}^{[n]}$ of X_{t-1}, the *State Transition Model* is applied by sampling the following distribution, using the *Control Vector* u_t as a parameter:

$$p(x_t|u_t, x_{t-1}^{[n]})$$

This sample, $x'^{[n]}_t$, is part of a new particle distribution X'_t.

The Particle Filter - Part 2

Computing the Weights:

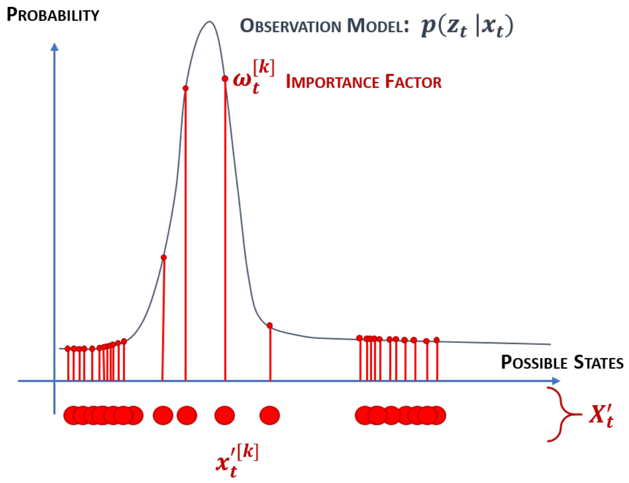

Probability
Observation Model: $p(z_t|x_t)$
$\omega_t^{[k]}$ Importance Factor
Possible States
X'_t
$x'^{[k]}_t$

Figure 2.7: Computing the Importance Factors Using the Observation Model

For each new particle $x'^{[n]}_t$, a weight w^n_t is computed. This *Importance Factor* reflects how likely the measurement is for a given state.

$$w^n_t = p(z_t|x'^{[n]}_t)$$

The pairs $(x'^{[n]}_t, w^n_t)_{n\in[\![1..N]\!]}$ represent the approximation of the *Posterior State Distribution.*

Resampling:

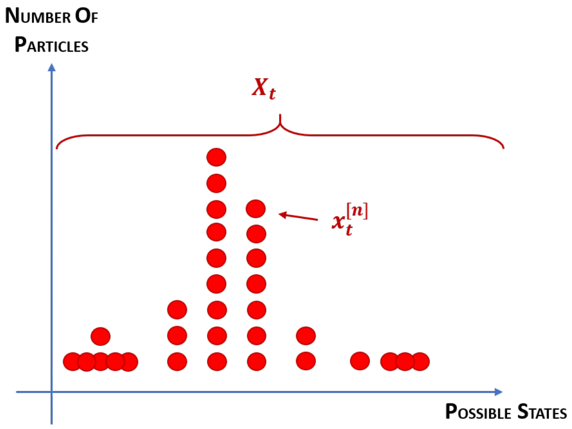


Figure 2.8: Posterior Particle Distribution After Resampling

The last step consists in drawing N particles with replacement from X'_t, using the previously computed weights as the probability of drawing a given particle. As a result, the new particle pool X_t, is populated by many redundant particles that had a heavy weight and less particles with lower ones. Most of the unlikely particles disappear during the resampling process.

2.2.3 Acoustic Navigation

In order to keep a bounded uncertainty about the vehicle location at any time, it is necessary to rely on external references as the INS drift over time. GNSS measurements coupled with the INS using sensor fusion algorithm can mitigate this drift issue. Unfortunately, as radio signals do not penetrate water, the AUV has to surface in order to perform a GNSS measurement.

Fortunately, it is possible to deploy underwater localization systems that rely on the exchange of acoustic signals between the AUV and on field instruments. This is the aim of *Acoustic Navigation*. The most widely deployed strategies for locating an AUV generaly fall within one of these three categories: the Long Baseline (LBL), the Short Baseline (SBL) or the Ultra Short Baseline (USBL). Locating a swarm of AUV introduces additional constraints that are addressed by more modern strategies such as One-Way-Travel-Time (OWTT) Navigation or Cooperative Navigation.

Long Baseline (LBL)
A Long Baseline systems consist in a network of acoustic transponders deployed on the seafloor and separated by a few hundred meters. Periodically, the AUV emits a pulse and the beacons in its vicinity send a signal as a response. The location of the AUV relatively to the beacon grid is computed using multilateration. It consists in measuring the different time-of-flights from the pings sent by the network of transponders. In terms of precision, LBL systems can provide very accurate measurements with a margin of error around a few centimeters. Unfortunately, operating a LBL system is heavy demanding. First, deploying several beacons on the seafloor and calibrating the network is complex. Second, the AUV can operate only within reach of the acoustic transponders, which limits the range of operations to the volume covered by the network. Last but not least, it is necessary to retrieve the underwater beacons after operation which make the LBL logistically demanding.

In the original LBL design, fours beacons have to be in range while the AUV emits one ping. Several improvements have been suggested in order to reduce the number of necessary transponders to one. Instead, the AUV emits a series of pulse and deduces its location by combining the different range measurements provided by a single beacon. The *Synthetic LBL* suggested by [58] rely on the fusion process between successive range measurements and an accurate INS including a DVL. Later, [8] proposes a similar method more fitted for low-cost sensors which does not require a DVL.

Furthermore, LBL acoustic beacons can be used in a Simulatenous Localization and Mapping scheme. An early use of a robust SLAM algorithm for underwater robotics have been presented in [72]. In this article, the AUV operates in a region covered by a network of LBL beacons with no *a priori* knowledge about their locations. An interesting insight of this study is the ability of the suggested SLAM algorithm to address the observation noise and outliers while using only range-only measurements. A notable derivative of the LBL concept consists in deploying buoys equipped with GNSS receivers [4]. In this configuration called *Inverted LBL*, the buoys communicate with the AUV by using acoustic transducers. Operating an Inverted LBL system is less tedious than a traditional LBL one as less efforts are required for calibrating the buoys array and retrieving the beacons.

Short Baseline (SBL)
Compared to the LBL, a Short Baseline is made of a small group of acoustic transducers separated by a few meters only. They can either be deployed underwater near a floating central station or directly to the hull of a surface ship. The AUV is equipped with a transponder that repply to acoustic ping emitted by the baseline. For a given ping returned by the AUV, different times of arrival are recorded by the baseline transducers. These differences are used by the central station to compute an estimate of the range and the bearing of the received ping, thus locating the AUV. A SBL system is less accurate than a LBL, however, it can be shown that the greater the baseline, the more accurate the SBL is [25].

Ultra Short Baseline (USBL)
An USBL system is made of two different units. The first one is an array of transducers coupled with a transceiver and the second one is a transponder. There are several ways to deploy each component. If, for instance, the AUV needs to be located by a surface vessel, the transponder is attached to the underwater vehicle while the array is positioned under the ship. Conversely, the configuration could be inverted in order to allow the AUV to track a reference beacon (a GNSS floating buoys for instance). Like in the case of the SBL, an USBL system performs a range and bearing measurement. The range to the transponder is computed by estimating the two-way travel time of a ping send by the transceiver like for the SBL. However, the bearing is derived by measuring an acoustic phase difference between the different array's transceivers. The main strengths of the USBL are its deployability and compacity. Indeed, a single instrument is needed and no extensive preparatory work is required before performing a survey.

Compared to a LBL navigation system, the localization estimate is more impacted by the change of the physical properties of water such as pressure or temperature.Indeed, they directly influence the sound wave propagation which may result in a less accurate measurement. However, when coupled with Inertial Navigation and an appropriate fusion algorithm, the USBL is a very efficient tool for underwater exploration. In [86], an USBL system is used as an aid for locating an AUV and thus georeferencing its sonar acquisitions. In order to address the measurement noise and the model non-linearities, a Particle Filter is used for fusing the information acquired by the different sensors. USBL are also used while operating towfishs [19]. Indeed, due to the flexible nature of the link between the sonar system and the towing surface ship, there is a high uncertainty about the payload location. Even if the towfish is usually equipped with an INS, the mechanical constraints that apply to the systems are difficult to model and negatively impact the localisation process. Consequently, an USBL is a tool of choice for improving gerorefencing data in various contexts of underwater surveys.

One-Way Travel Time Acoustic Navigation (OWTT)

The previous systems (LBL, SBL and USBL) have played a proeminent part as acoustic localization systems during the past decades. Unfortunately they suffer from limitations that make them inefficient for some modern underwater applications. One of their disadvantages is that they are not optimal for locating a swarm of AUVs. Some complex underwater missions like mine hunting require to operate a group of heterogenous AUVs. As a result, a need for swarm-efficient underwater acoustic positioning system have arisen.

In order to monitor several AUVs using traditional acoustic navigation techniques, it is necessary to use a Time Division Multiple Access (TDMA) approach which dramatically limit sthe localisation update frequency per AUV. More modern design of LBL systems involving a wideband acoustic spectrum enable swarm applications. Unfortunately, all the other constraints related to the use of LBL beacons remain.

A decade ago, techniques based on Single Beacon One-way Travel Time measurements have been implemented and tested underwater [31]. A single beacon equipped with an acoustic modem periodically emits data packets to the AUV swarm. As each vehicle inboard clock has been synchronized with the beacon the time-of-flight of the packet is enough for estimating the slant range towards the beacon. Furthermore, the packet contains information itself, regarding the absolute location of the beacon for instance. This approach allows each AUV to concurrently update their location estimate accordingly. In order to extend the range of operations, the beacon can be mounted over a surface ship that may move. Fusing information between the swarm and the beacon is a challenging task, especially due to the decentralize nature of the estimation process. In [103], an approach relying on the use of a Centralized Kalman Filter that concurrently use data from the AUV and the ship is used as a post-operation processing. All sensor data is used simultaneously to compute the best estimate possible of a state vector that encompasses variables related to the ship and the AUV, thus improving the *a posteriori* localization of the underwater vehicles. More recent implementations of this OWTT navigation aim to make it efficient for AUVs swarm equipped with low cost INS for instance [26].

Cooperative Navigation

The group of techniques that apply to a swarm of AUVs that exchange information in order to increase their knowledge about their location and their environment fall into the Cooperative Navigation category. The key of such approaches relies on the use of acoustic modem. OWTT techniques can be applied within a swarm of AUVs without relying on a surface reference ship for example. The foundation of cooperative navigation for swarms of aerial or land autonomous systems originated in the mid-90s.

Unfortunately, the underwater environment makes communications highly limited and challenging. Acoustic modems can send a few hundred of bytes per seconds over a few kilometers (<5km). Due to the slow propagation of sound compared to radio waves, underwater communications also suffer from very high latency. Last but not least, Frequency Division Multiple Access (FDMA) is also limited by a narrow band ($\sim$ 15 KHz) for communicating between multiple agents. As a result, both range and volume of data are limited by the physical properties on the underwater environment. Among the earliest implementations of cooperative navigation, the approach presented in [7] was relying on the use of a *Leader AUV* sharing navigation information with a group of followers equipped with observation tools for scanning the area. The leader agent is devoted to navigation purposes as its is the only one equipped with high precision INS. Its estimates are communicated to its followers using acoustic modems. Later, [9] presents an experiment where a swarm exchanges data and perform range measurement without relying on a leader or a surface baseline. More recent studies involve cooperative SLAM. In that case, information about the local map generated by each AUV is communicated to its teammates. However, as the amount of information that can be transmitted to each other is limited, it is necessary to limit the communication to the most relevant information. [75] presents an implementation of a Graph SLAM framework were the exchanged information between the different agents is optimized in order to meet the communication channel limitations in the underwater environment.

2.2.4 Geophysical Navigation

2.2.4.1 Principle and Key Instruments

Rather than relying on artificial beacons, like in the case of acoustic navigation, it is possible to probe the environment in the search of natural reference points. This is the aim of *Geophysical Navigation*. In order to bound the navigation uncertainty due to the INS drift, the AUV relies on a map of the environment. Acquisitions from sensors dedicated to the observation of the vehicle surroundings are processed in a search of matching reference points within the map.

Of course, the nature of the map depends of the type of available sensors. For instance, an AUV carrying a Multi-Beams Echo Sounder requires a Bathymetric Map of the region. It is important to note that there are two different initial set-up: either the AUV has a prior knowledge of the scene and carries a pre-existing map, **or**, the vehicle operates in uncharted water. In that case, it has to simultaneously generate its own map while using it for addressing the localization problem. This initial condition is critical as it dramatically impacts the way geophysical navigation is implemented.

The main advantage of Geophysical Navigation is that it is self-contained. As it can bound the INS drift, it is the perfect aid for improving underwater navigation in unknown regions. Developing robust Geophysical Navigation may enable a wide range of new applications requiring autonomous agents operating on their own. As a result, this is an intense and complex field of research.

Different Technologies and Sensors

Geophysical Navigation may rely on a large variety of instruments that use different types of technologies. Generally speaking, observation measurements can be either performed using either acoustic, optical or magnetic signals:

- ***Magnetic and Gravitational Sensors:***
  The Earth Magnetic and Gravitational fields are valuable sources of information that can be used for AUV localisation. Using either a magnetometer or a gravimeter, it is possible to measure the local anomaly and use a reference map in a similar fashion to Terrain Based Navigation relying on Bathymetric maps. As these sensors are silent (no acoustic pulse emitted) and self-contained, they have a very high potential for military applications that require stealth. However, studies about the operational efficiency of such techniques are still limited.

- ***Optical Sensors***: Underwater cameras can be deployed to acquire either video or high definition images. They can be used for object detection, identification and tracking. Most of the techniques developed in the field of *Visual Odometry* can be directly applied to such cameras, including many image processing techniques like keypoints detectors and extractors frequently used in robotics. However, compared to their use for land or aerial vehicles, their range is limited due to the high absorption of the light by water (no more red color after 20 meters of depth for instance). There are suitable tools for observing objects located at a close range (hull inspecting, man-made structure monitoring, etc...). Stereo cameras can be deployed for producing 3D images. Compared to acoustic instruments cameras are cheap which make them very interesting for low-cost AUVs operating in shallow waters.

- ***Acoustic Sensors:*** SOund Navigation And Ranging (SONAR) systems encompass all the instruments that rely on the propagation of acoustic signal for performing diverse tasks including range measurement, imagery or communication. They can be divided into two categories of interest: *Passive Sensors* that "listen" to the environment and *Active Sensors*. The later category is used for remote sensing applications. They rely on the emission on an acoustic pulse and measure the returned echo. Among active sonars, *Sonar Imagers* measure the intensity of the back-scattered pulse in order to create an Intensity image. Their high resolution, range and sensitivity to the type of floor make them very useful for studying and mapping the underwater seafloor. With regards to the needs of Geophysical Navigation, acoustic image processing is challenging due to the specificities of the Slant Range geometry of observation and the underwater environment (multipath, attenuation, shadows…). *Ranging Sonars* on the other hand are used for measuring the distance between the seafloor and the payload. They are generally deployed for performing bathymetric surveys or for *Terrain Based Navigation* (TBN).

2.2.4.2 Focus on Active Sonars

Sonar Imagers :

- **Side Scan Sonar (SSS)**
  Thee SSS is a lateral observation sensor that forms two images by emitting an acoustic pulse on each side of the payload, perpendicularly to the heading. By measuring the intensity of the backscattered signal, the SSS creates a waterfall image of the seabed. The frequency of the ping emission defines the maximum range which is usually around 50 $\sim$ 100 meters on each side of the vehicle. Another limitation of the range is due to the attenuation of the sound in water that depends of the frequency of the emitted pulse. Higher frequencies mean higher resolution, but also shorter ranges. The SSS offers a good trade-off between cost and operational efficiency. It exceeds at quickly mapping large regions of the seabed. The main downside of the SSS is its non-constant resolution which introduces uncertainty for image mosaicking and analysis.

- **Synthetic Aperture Sonar (SAS)**
  The SAS can be seen as a modern evolution of the SSS inspired from its radar counterpart, the Synthetic Aperture Radar. This instrument requires a high processing power in order to generate an image with an along-track constant resolution. By far the SAS is the most accurate tool for scanning the seafloor. Unfortunately, there are some operational downsides. First, it is an expensive instruments that requires more inboard computational power and need more power. Second, it requires to operate at a lower speed compared to the SSS while acquiring.

- **Forward Looking Sonar (FLS)**
  This instrument, generally mounted on the front of the vehicle, generates a 3D image of the forward direction covered by the sensor range. Generally used for collision avoidance or fishing, this sonar can be also used for filling gaps in the *nadir region* for lateral sensors.

Ranging Sonars :

- **Altimeter Single Beam**
  The altimeter measures the distance between the AUV and the seafloor using a single narrow acoustic beam. Due to the low area covered by such instruments, it is necessary to use a series of measurements for disambiguation while using a bathymetric reference map. Fortunately, it is an affordable instrument compared to more sophisticated bathymetric sensors.

- **Doppler Velocity Log (DVL)**
  The main use of a DVL is to provide a ground speed measurement as discussed in 2.2.1, however, it can also performs depth measurements analogously to an altimeter. Due to its ability to both work with INS and TBN, DVL became very popular, even for low-cost AUV designs.

- **Multi-Beam Echo Sounder (MBE)**
  In order to perform a measurement over a large swath instead of a single point bellow the vehicle, Multi-Beam Echo Sounders use beam forming arrays that simultaneously ensonifies a series of points in the across-track direction using a single acoustic pulse. A second transducer is used for collecting the backscattered signals and performs a time-of-flight measurement. After assembling the different echoes, it is possible to produce a terrain elevation of the swath. MBEs are the most accurate and efficient tools for producing large-scale bathymetric surveys. They also are more efficient for Terrain Based Navigation as a single swath acquisition can address the disambiguation problem while using the reference map. Analogously to the SAS, they are massive, expensive and require heavy inboard resources which may make them not suitable for some AUV designs.

2.2.4.3 Examples of Geophysical Navigation

There has been several implementations of Geophysical Navigation in the underwater environment so far. The oldest one is *Terrain Based Navigation*. This approach has been inspired by the navigation algorithms that were deployed on missiles for reaching their target using an elevation reference map.

Two main strategies have emerged in the past, namely TERCOM (Terrain Contour Matching) and SITAN (Sandia Inertial Terrain Aided Navigation) and have been successfully applied to the use of bathymetric maps in conjunction with ranging sonar systems. Both strategies rely on applying matching functions between the acquisition and the reference map in order to update the navigation filter accordingly. The main difference between these two strategies lies in the fact TERCOM computes the best match between a set of acquisitions and the reference map before incorporating this match as a position fix/measurement in the *Update Phase* of an estimation filter. On the other hand, SITAN works in a sequential fashion as every bathymetric measurement is used for updating the filter. Other approaches derived from image registration techniques like ICCP (Iterative Closest Contour Point) have been developed later [50], notably in the case of gravitational reference maps.

In fact, Terrain Based Navigation strategies can also be applied to other types of modalities including geomagnetic or gravitational reference maps. However, such type of applications are more rarer than bathymetric TBN in the literature. Quite recently, [107] presented a gravity aided navigation using a Point Mass Filter, another type of non-parametric estimation algorithm for estimating the localization of an AUV using a local gravitational anomaly reference map. One strength of TBN-related methods is that they rely on maps that generally offer a high degree of variability, thus enabling efficient localization. A recent review of TBN approaches applied to the underwater environment is presented in [64].

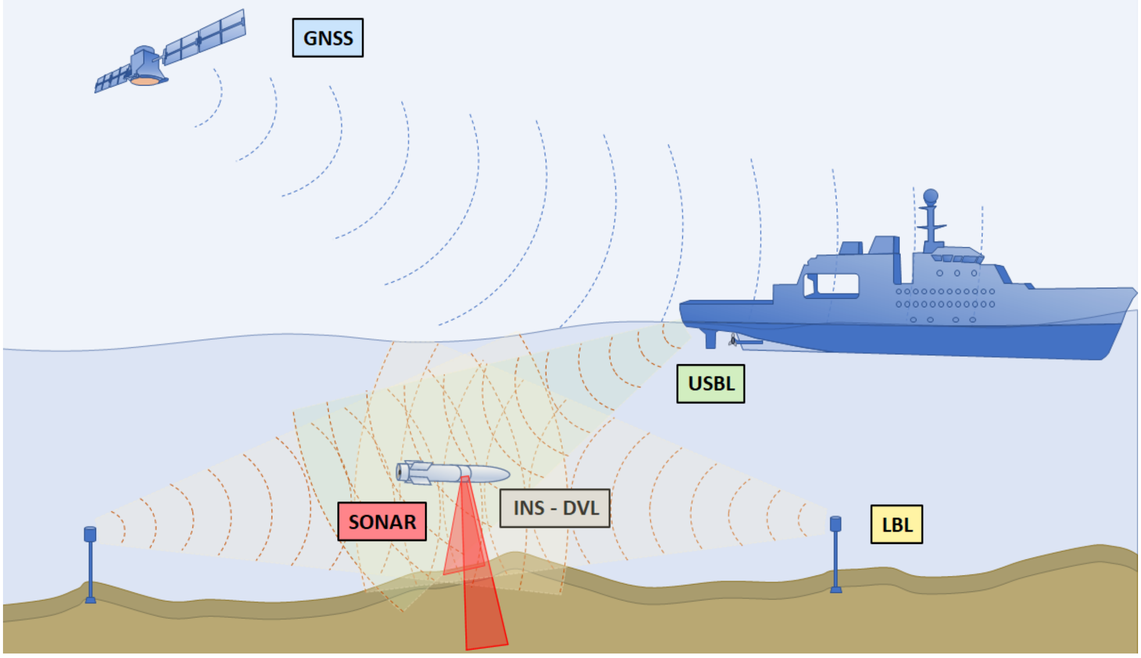


Figure 2.9: Illustration of different Underwater Navigation Systems.

The limitation of TBN lies in its requirement of a reference map. As a result, developing techniques that would enable fully autonomous navigation without the need of prior knowledge about the operational area is critical for enhancing AUVs capabilities. This is precisely the aim of Simultaneous Localization and Mapping (SLAM).

SLAM algorithms can be seen as an evolution of many of the fusion algorithms used for estimating the trajectory, on the difference that they also estimate the map of the environment according to the set of geophysical observations and motion measurements. There are different types of SLAM, depending on the way the map is represented. This approach introduces numerous additional challenges that are strongly sensor-driven. Early implementations of underwater SLAM were relying on the use of artificial features, easily detectable [105]. The current challenge related to SLAM is the ability to be efficient using natural features only, taking into account the AUV's inboard capabilities. The review of underwater SLAM is presented in Chapter 4, after discussing the specificities of Side Scan Sonar Imagery in Chapter 3.

Other approaches that do not rely on either TBN or SLAM do also exist. For instance, [74] uses video tracking in order to follow telecommunications cables in order to inspect them. As a summary, Figure 2.9 presents most of the instruments for underwater navigation that have been discussed in the previous sections, including this one.

2.3 Towards more constraining operational scenarios

2.3.1 Towards more autonomy

Autonomous Underwater Vehicles have been a very intense field of research during the past 20 years. In the late 90's, early deployments of the first generation of AUVs was discussed in the literature [94]. However, their capabilities were limited by different factors [39]. First of all, computational intensive task were impossible due to the limitations of the inboard hardware. As a result, many applications requiring heavy online data processing were challenging. Second, their range of operability was strongly limited by different factors. Power management was one of them, but the most critical issue was addressing the navigation problem. Indeed, high-grade INS are still bulky and expensive nowadays. As a result, early AUVs designs required significant logistical support, generally using surface ships, or a network of acoustic beacons.

Improving the autonomy of underwater technology would strongly benefit a wide range of applications including both civilian and military uses. For instance achieving low-cost precise navigation of autonomous agents would facilitate underwater surveys. Indeed, if the sensor location could be bounded under an acceptable level, it would be possible to quickly georeference remote sensing acquisitions, thus enabling more efficient surveys. For example, in the field of bathymetric surveys, the S-44 norms defined by the International Hydrographic Organization (IHO) [73] details operational requirements in order to ensure quality survey. It states that the maximal horizontal sensor location uncertainty should remain under two meters to comply with the norm and ensure the highest accuracy. As discuss in [70] with an experimental survey using a Daurade AUV, the operational duration under such requirement could be met is heavily dependant on the INS quality. As suggested by the authors, more advanced navigation approaches like SLAM would compensate the need of expensive IMUs, thus reducing the costs of the survey. The need of automating maintenance and monitoring tasks especially on offshore structures motivates the deployment of more and more autonomous units.

Some modern applications of AUVs tend to be efficiently addressed by swarm strategies. Deploying multiple AUV at the same time may strongly reduce the time of operation for performing tedious and time-consuming surveys of very large areas. A swarm of heterogeneous units may be critical for executing complex tasks like Mine Countermeasure (MCM)). The complementarity of different instruments for detecting, tracking and analyzing potential threats is the key for optimal results. For instance, an AUV equipped with a lateral sonar can quickly screen a large area in the search of suspicious echoes. Potential threats location can be transmitted to another unit carrying a high-definition Forward Looking Sonar for object identification and classification.

The military environment also introduces new constraints and challenges related to a greater need for truly autonomous systems. Assuming that a lone AUV has to perform a high risk recon survey in potentially hostile waters, reducing the vehicle signature becomes a high priority. First, as the AUV could not surface in order to acquire a GNSS signal, it has to rely on inertial and possibly geophysical navigation. Even for military grade INS, long duration operations still requires to performs localization corrections. As a result, SLAM-based navigation strategies constitutes the best approach as long as the AUV can identify relevant natural landmarks. Another interesting operational constraint in mission requiring stealth is the acoustic signature of active sonar used for probing the environment. When acquiring geophysical data, the AUV emits acoustic pulse that can be detected by a potential threat from afar. Using inboard passive sonars, it might be possible to detect incoming foes and enter a 'silent mode' to avoid being detected. As a result, geophysical navigation would stop for a time and the AUV should rely uniquely on INS. The AUV could also stop its engine and let itself drift for a while to remain completely silent. This would introduce a challenging operational scenario.

Improving Operational Resiliency is one of the most critical topic for ensuring autonomy in more and more complex scenarios. The diversity of possible failures that may occur and affect different instruments have to be taken into account in advanced navigation strategies.

During the last decade, AUVs had a major advance in terms of operational efficiency, mostly due to the deployment of more important inboard computational power, enabling more complex in-mission analysis. Current highly active fields of research like Artificial Intelligence strongly benefits AUV's capabilities notably in analyzing their environment and in object classification and detection (MCM for example).

2.3.2 Addressing the current limitations of Geophysical Navigation

The main challenge for enabling efficient geophysical navigation under a SLAM scheme is the unstructured nature of the underwater environment. The key for efficiently running a SLAM algorithm is to build a map and search for correlations between the map and the live acquisitions in an efficient way. Building a map in an environment with low diversity or few characteristic features is complex. Any algorithm looking for correlations is prone to potential data association errors or may face a lack of relevant information. For instance, a bathymetric map generated in a flat region is useless for locating the AUV using a SLAM algorithm. On the other hand, many seafloor objects may have very similar signature which make the task of finding ***closing loop observations*** highly difficult.

The physical properties of the active sensor may also introduce additional challenges for identifying and characterizing reference points. For instance, in the case of lateral sonars, like the SSS or the SAS, the Slant Range geometry of observation introduces strong geometrical deformations that alter the shapes of objects. The presence of shadows also introduce additional uncertainty. Furthermore, lateral observation forces the AUV to move in a sequence of lines in order to perform a good acquisition. Consequently, a same object is spotted only a couple of times during a survey, when two sonar images overlaps. As the two observations are made under two different points of view, the signatures may dramatically differ, making landmark association highly non-trivial.

As a result, it is important to adopt robust SLAM strategy that can handle outliers in an efficient way. Several approaches in Sonar Image Registration can be applied for finding correspondences between two acquisitions, however, it is critical to make sure that the retained methodology could cope with a high degree of localization uncertainty especially when advanced operational scenarios involving data interruption are faced.

In fact, the future of underwater navigation will rely on the use of heterogeneous sensors with complementary modalities in order to run robust SLAM algorithms without the need of high-cost IMUs.

2.3.3 Objectives of our research

We works towards designing a framework that address different challenges related to underwater navigation while using a Side Scan Sonar and an INS in a highly constrained environment.

Among the objectives of this thesis, we aim to:

- Discuss methodologies for detecting and matching reference points using sonar images under a global registration problem
- Address scenarios where motion measurements are interrupted
- Develop a robust strategy for detecting possible erroneous data associations using a robust SLAM algorithm and an *a posteriori* scene validation relying on environment analysis

The following Chapter introduces the challenges of Side Scan Sonar data processing under a global registration problem.

Chapter 3

Underwater Mapping with a Side Scan Sonar

Highlights of the Chapter

- Active sonar systems form a group of remote sensing instruments that are particularly useful and efficient for underwater exploration and mapping. The categories of imagers such as the Side Scan Sonar (SSS) or Synthetic Aperture Sonar (SAS) excel in sedimentary analysis and enable the fast generation of large mosaics. They play a key part in the fight against underwater mines due to their long range, high sensitivity and quick acquisition speed.

- This efficiency makes them suitable candidates for an autonomous system that has to map an uncharted area. However, the specificities of the underwater environment, coupled with the properties of the lateral active sonars cause some difficulties in the registration of the acquisitions, especially when the uncertainty on the global location of the system becomes significant. Indeed, unlike in the case of a city or an indoor place, the marine environment shows little to no structure. As a result, the sonar may produce hardly discriminating visual signatures of seabed features. Furthermore, it is also an evolving environment. Due to water currents and human activity, some natural landmarks may spontaneously disappear or change in appearance over time.

- Finding remarkable landmarks is a complex task, if any exists at all. In general, in computer vision, this implies a multi-stage process: detecting, characterizing and associating. Historically, there are different types of methodology for registering sonar acquisitions. The so-called symbolic approaches are based on the characterization of salient features of the environment such as echo/shadow complexes or the analysis of textures. The iconic methods are more oriented towards signal analysis and use the similarities of the underlying probability distributions of some regions of the mosaic. There are limits to these techniques and sometimes a trade-off between speed of analysis and accuracy is necessary. These approaches are particularly challenged in the context of global registration.

- That is why the implementation of underwater navigation based on a SLAM algorithm is complex. Generally an accurate inertial unit, although drifting inexorably, is sufficient to limit the problem of registering two sonar acquisitions to a few meters of uncertainty (or tens of meters). However, global repositioning is a much more complex problem because the search range can be much larger. From an operational point of view, this can require too much computation and even compromise the relocation process. This problem can occur when there is an unexpected interruption of the acquisitions, in particular after a system failure.

- This chapter focuses on the processing of sonar acquisitions in order to perform data associations for the SLAM algorithm. First, the steps required to form a sonar mosaic are reminded. As opposed to an optical camera, an image is not immediately available. For each acoustic ping emitted by the active sonar, the intensity of the received backscattered signal is measured. This signal is sampled at regular time intervals and is presented in the form of a series of Bins which are used to calculate the values of the pixels of the mosaic. Several treatments are then necessary to get to a georeferenced product. These steps introduce approximations that affect the fidelity of the representation of the environment. In the case of SSS for example, the pixel resolution is non-uniform. Moreover, the SNR degrades quite rapidly at the end of the range. Finally, sonar imagers do not have elevation resolution. Without an accurate bathymetric survey, one is forced to create a 2D mosaic, which introduces a significant bias. This can therefore lead to a misinterpretation of the environment that may compromise the map built by the autonomous system.

- In this section, we present the work that has been carried out to extract and associate images from disjointed acquisitions, seeking a compromise between efficiency and accuracy. These associations then serve as initial hypotheses for the SLAM algorithm in the form of what will later be called Loop-Closures Constraints. Because of the factors listed above, a number of associations are erroneous. It will then be up to the SLAM algorithm to fuse the sonar and motion data to discard these outliers. The operational complication, which is the key point of the scenario addressed by this research work, strongly influences the ability of the SLAM algorithm to filter out these errors. In addition to the data association phase, image processing will also play an a posteriori role in assessing the plausibility of the map after the SLAM session. To this end, some methods complementary to those used to form loop-closures constraints are applied. It is then a matter of detecting inconsistencies in the merged mosaic that may result from errors in associations formed before the SLAM session.

3.1 Presentation of the Side Scan Sonar

3.1.1 Overview

The **Side Scan Sonar** (SSS) is an **active acoustic sensor** that has been widely used for seabed observation since the early 1960s. It consists in an array of transducers that emits **acoustic pulses called Pings** at regular time intervals in the **across-track direction** of the vehicle. The seafloor acts as a receiver and generates a **backscattered wave** that is captured by the sonar transducer. This signal is called the **Echo**. Using beamforming processing, the raw measurement is stored into a series of bins that contains the backscattered acoustic intensity over the whole ensonified region.

Because of the spherical propagation of the acoustic wave, the seafloor ensonified region looks like a conic-shaped **swath** as illustrated in 3.1. One of the main properties of the SSS is that there are two different resolutions to take into account. First, due to the spherical propagation of the acoustic wave, the **across-track** resolution (or range resolution) is inversely proportional to the **Slant Range**. Basically, the pixels get thinner far from the sensor in the direction orthogonal to the vehicle motion. Conversely, because of the antenna opening, the **along-track** resolution gets worse when moving far from the Nadir. In that case, the pixels get larger in the direction parallel to the vehicle heading in the far field. This is described in Figure 3.1, detail number (6). This later weakness is corrected by one of the SSS evolution, the Synthetic Aperture Sonar. By combining several overlapping pings, it is possible to achieve a constant along-track resolution. Unfortunately, this is a rarer sensor as it is dramatically more expensive and computational-demanding.

For example, in the case of the sensor that has been used for generating the dataset of sonar images presented in our research, a Klein 5400, the resolutions are as follow [1]: Along-Track resolution is 10cm at 38m and 20cm at 75m, while Across-Track is 3.75cm before the Slant-Range correction step [Approx. 12cm near Nadir for 3.8cm at 75m for a sensor altitude of 10m].

Another significant limitation of SSS is that it has **no resolution in elevation**. Either a prior map of the local topography is available or the flat bottom hypothesis has to be made in order to be able to project the **Slant Range** measurements over a raster. This last approximation could produce significant error if the ensonified area is located in a slope. Unfortunately, there is no way to correct this issue if the SSS acquisitions are the only available data. An interesting feature of the SSS is that large objects laying on the seabed with a sufficiently high height cast a shadow on the sonar swath. While there is no information available about the dark side of the object, the shape of the shadow could be processed to extract some characteristics and could be used as a source of descriptors for data association and navigation. It is important to mention that the shape of the shadow dramatically changes with the **Grazing Angle.** [Note: It is basically the angle between the Nadir and the center of each bin: Figure 3.1 presents an example between lines (3) and (4).]

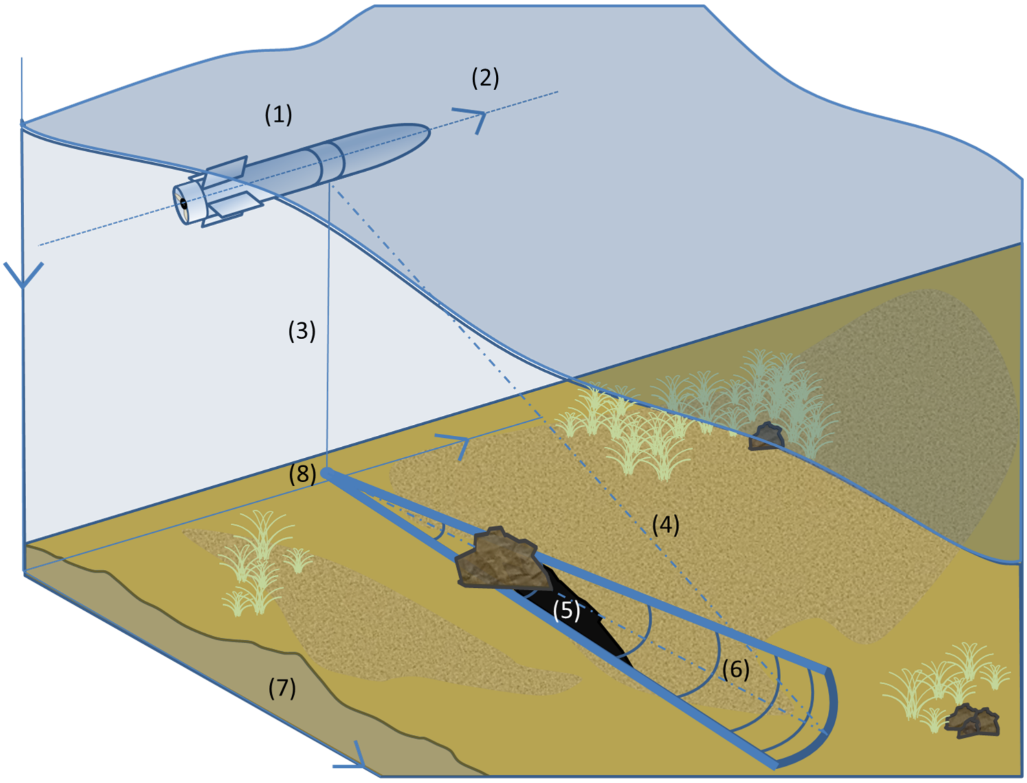


Figure 3.1: AUV Observing the Seabed Using a Side Scan Sonar
(1): AUV, (2): Heading, (3): Sensor Altitude Relative to the Seabed,
(4): Across-Track Slant Range (Direction of Observation),
(5): Acoustic Shadow, (6): A Single Acoustic Ping Bin,
Note: The Across-Track Resolution Improves with Distance,
(7): Seabed Elevation, (8): Nadir

Because of the design of this type of Sonar, it is necessary to maintain a rather straight trajectory when mapping a region. Traditionally, the vehicle observes a "lawnmower" pattern as presented in Figure 3.2 to avoid creating large gaps in the acquisition because of frequent turns. Before the development of AUVs, SSS were mounted on towed fishes, torpedo-looking submerged platforms, that are towed by a surface ship using a flexible cable. As the exact localization of the sonar is not directly measurable, an Ultra-Short Baseline is used to locate the fish relatively to the boat. In this scenario, it is possible to use the ship GPS coordinates to georeference the acquisitions directly. In the case of an AUV, it may have to surface time to time to refine its current location as inertial navigation data is not accurate enough to perform automatic mosaicking.

SSS acquisitions are great for analyzing the seafloor sediments and objects. Indeed, the intensity of the backscattered echoes strongly depends on the composition of the re-emitter (the object). It is therefore a useful tool for detecting and classifying different types of regions using their respective acoustic signatures. Another general property of a SSS is their operating frequency. Generally, they emit waves in the KHz bandwidth. The higher the frequency, the higher the image resolution. However, water absorption also increases with frequency. Consequently, it impacts the maximum range as the **Signal to Noise Ratio (SNR)** decreases faster.

Lastly, the maximum range is a free parameter that is tuned by the user prior the mission. It consists in setting a maximum Time of Flight (ToF) for each ping. Using the sound propagation velocity in water (on average $1500m/s$) and the ToF, it is possible to compute a maximum Slant Range (SR). Given the vehicle altitude and the flat-bottom hypothesis, it is possible to derive the maximum distance between the nadir and the last projected bin on the seafloor using Pythagoras's theorem:

For Example, if maximum $ToF = 0.1s$, Altitude $= 10m$, Sound Speed in Water $= 1500m/s$:

- Max Slant Range $= 75m$
- Max Projected Range$= 74.3m$
- *Note: The acoustic wave actually travels over 150m, as it has to do return journey after being backscattered.*

As the maximum range over the seafloor depends on the sensor altitude, it is important to maintain it at a relatively stable altitude and quite close to the bottom. The maximum ToF is generally constrained by the SNR of the image. In practise, SSS have a range comprised between between $50m$ and $150m$ on each side.

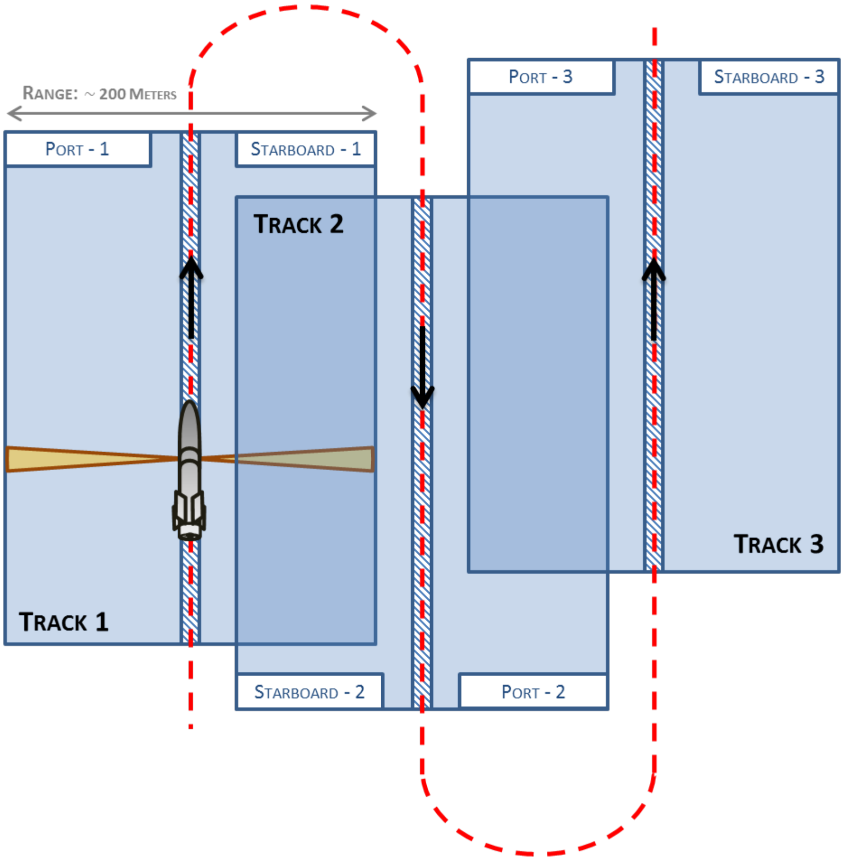


Figure 3.2: Standard Trajectory Pattern of a SSS Survey Mission

3.1.2 The Sonar Signal

An example of Sonar Echo is presented in Figure 3.3. The plot that lays bellow represents the backscattered intensity measured for each bin contained in Ping 6396. There are two important regions: the water column and the rest. The first part represents the time during which the sonar listen without measuring any signal: indeed, the wave propagate through water without encountering any obstacle. After a certain point it reaches the seafloor. This event is called *the first bottom return (FBR)*. By measuring the ToF associated to this point on both images, it is possible to get an estimate of the altitude (although it is not accurate).

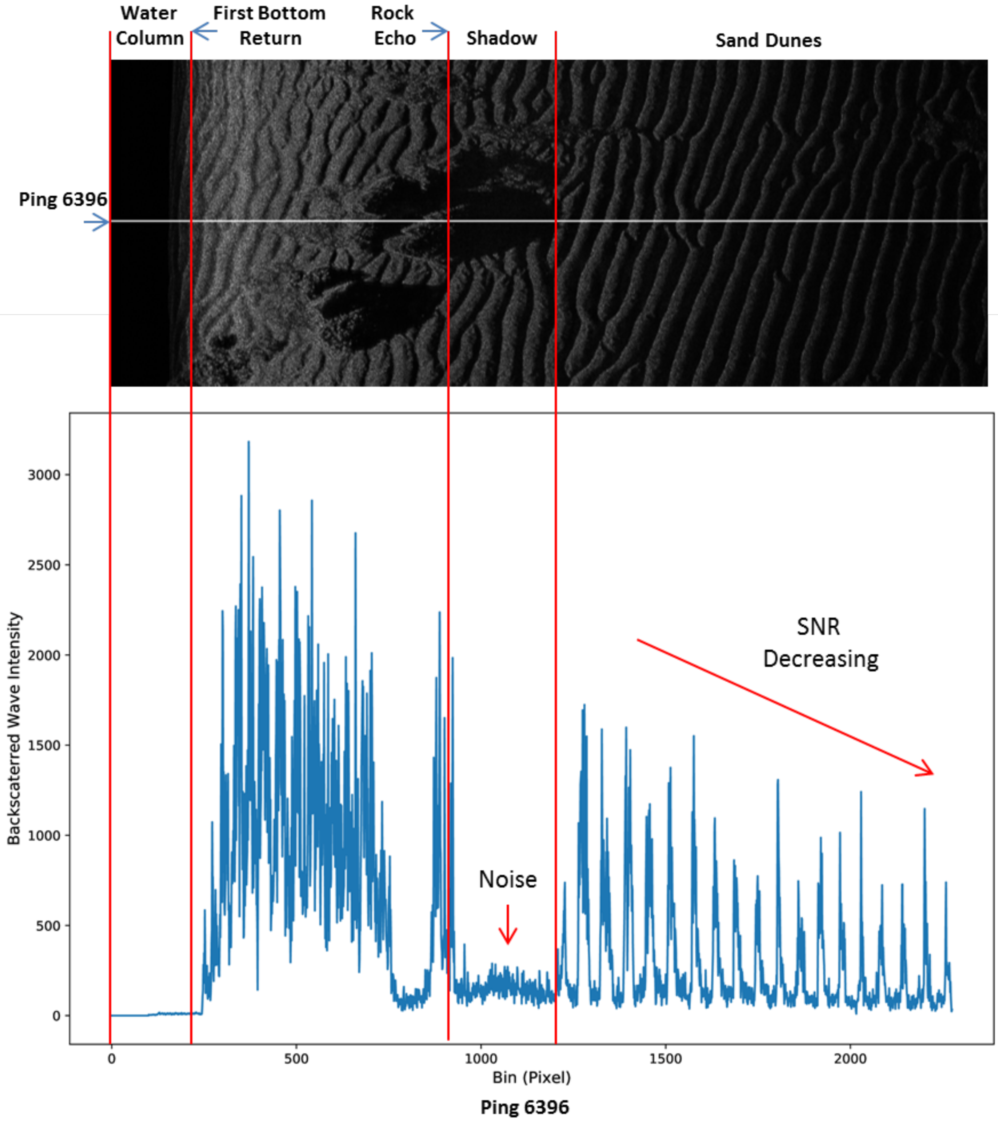


Figure 3.3: Measurement Following the Emission of Ping 6396

Beyond the FBR, it is important to notice that intensity is much more important close to the nadir region. This is due of the uneven ensonification resulting from the Slant Range geometry of observation. Propagation Loss plays a key part here as a backscattered wave intensity is inversely proportional to $d^4_{sonar-target} = (d^2_{Sonar\ to\ TargetSR} * d^2_{Target\ to\ SonarSR})$ as it assumed to be a spherical wave. (Note: in shallow waters, seafloor and sea surface act as a waveguide which limits the propagation loss. In that case a cylindrical wave model is more appropriate. However this is not used in SSS imagery as max ToF are quite limited). When producing an image, Transmission Loss is Generally compensated by the Time Varying Gain. While it prevents from having an image overly bright on one hand, it does not improve the SNR as this gain also increases the noise. Therefore detection becomes more and more challenging afar from nadir.

Figure 3.3 is an interesting example as it showcases the shadow of a seabed structure: after a local intensity maximum, a large range of bins are equal to the ambient noise. This feature is key for detection. On the later part of the plot, it is also possible to spot a succession of highs and lows that corresponds to a sand ripples region.

The raw sonar data is registered in a waterfall file as presented in Figure 3.4, an example extracted from our dataset. There are basically three critical steps in order to turn the raw data into an actual image (or mosaic). The first one is the intensity correction, that comprises the TVG step,. The second one is the slant range correction, using the flat bottom hypothesis. The last step consists in projecting each ping within a raster.

The TVG problem can be tackled by either using a heuristic [45] or deriving a more accurate model using the sonar antenna properties [56], [21], [17]. The later one may result from a theoretical reasoning, it is also very dependant on the knowledge of the antenna parameters and the vehicle pose. It is also a lot more time consuming. Furthermore, there is variety of other factors that impact the acoustic signal like absorption by water or the inhomogeneity of the propagation medium. Indeed, wave propagation is impacted by the water temperature, salinity or pressure. While in shallow waters these factors could play a negligible part, it becomes a more significant problem in deep ocean. Overall, the ensonification pattern on the seafloor could be impacted by a series of parameters that are difficult to estimate. Therefore, an heuristic correction may sometimes appear as a preferable option if the antenna parameters (opening, orientation etc ...) are not accurately known.

The Slant Range correction is straightforward under the flat bottom hypothesis. While it introduces a major bias in the environment representation, potentially altering some of the key features shapes, there is little to no alternative without athe addition of a bathymetric map. However, the last step of the image generation process raises many issues.

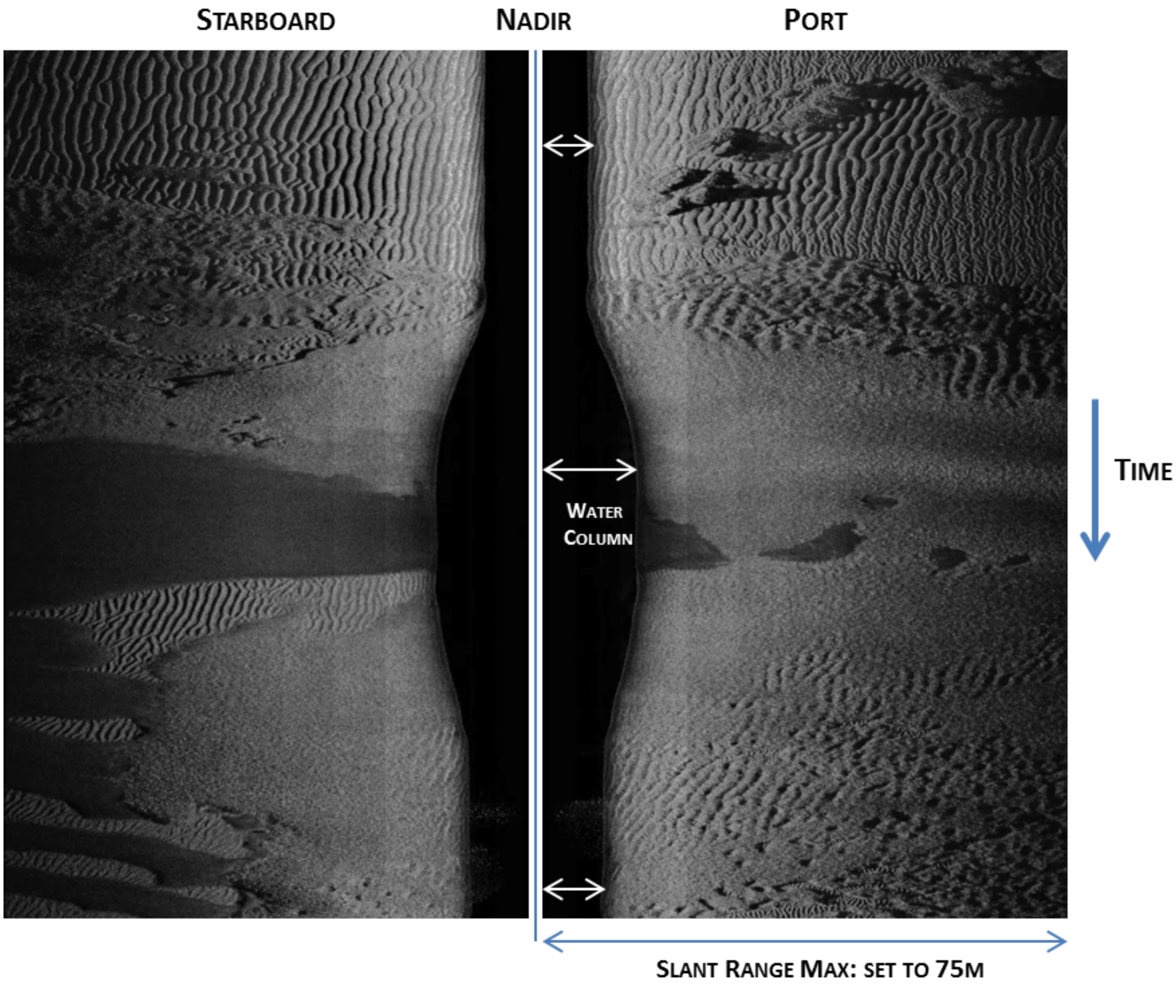


Figure 3.4: Example of Raw Side Scan Sonar Acquisition.
Successive Measurements are Stored in a Waterfall Datafile.
Further Processing Steps are Required to Generate a Proper Image.

In order to project the bins over a raster, the vehicle pose plays a critical role. The vehicle heading is necessary for defining the across-track direction, while the absolute position is critical for estimating the nadir location at the time of the ping emission. Consequently, if the estimated trajectory is inaccurate, the georeferencing of the sonar swath will be biased. This is the very reason why a post-acquisition registration step is required. Due to the increasing uncertainty resulting from the INS drift, the sonar images are slightly more and more inaccurately located over time. In order to solve this problem, the comparison of overlapping acquisitions is done via a local registration problem.

As highlighted in Figure 3.2, a standard survey involves a series of successive overlapping sonar images so similar features/patches could be matched. These matches can either be used as an observation measurement for the navigation algorithm to correct the whole trajectory estimate (this is what we would like to achieve in our research, but under a specific scenario) or the images could be directly warped to produce a more consistent representation without revisiting the sensor location. Another significant challenge is the non-uniform resolution of each bin as highlighted in Figure 3.1. It is possible to approximate the shape of the bins by rectangles of varying size, however it introduces another bias in the representation. Other approach like using a probability map [18] could be used to fuse different bins in order to achieve a more accurate representation. However it is also very computational intensive One has to keep in mind that on-board resources might be limited and performing long and tedious calculations could significantly affect the vehicle operational capabilities.

Figure 3.5 is an example of sonar image generated by the framework that we have developed. Here, the raw data processing was kept as simple as possible (Gain calculated via the Propagation Loss , Flat Bottom Hypothesis & Rectangle Bins Modelling). Regarding the navigation data, the trajectory is computed using the INS measurements only.

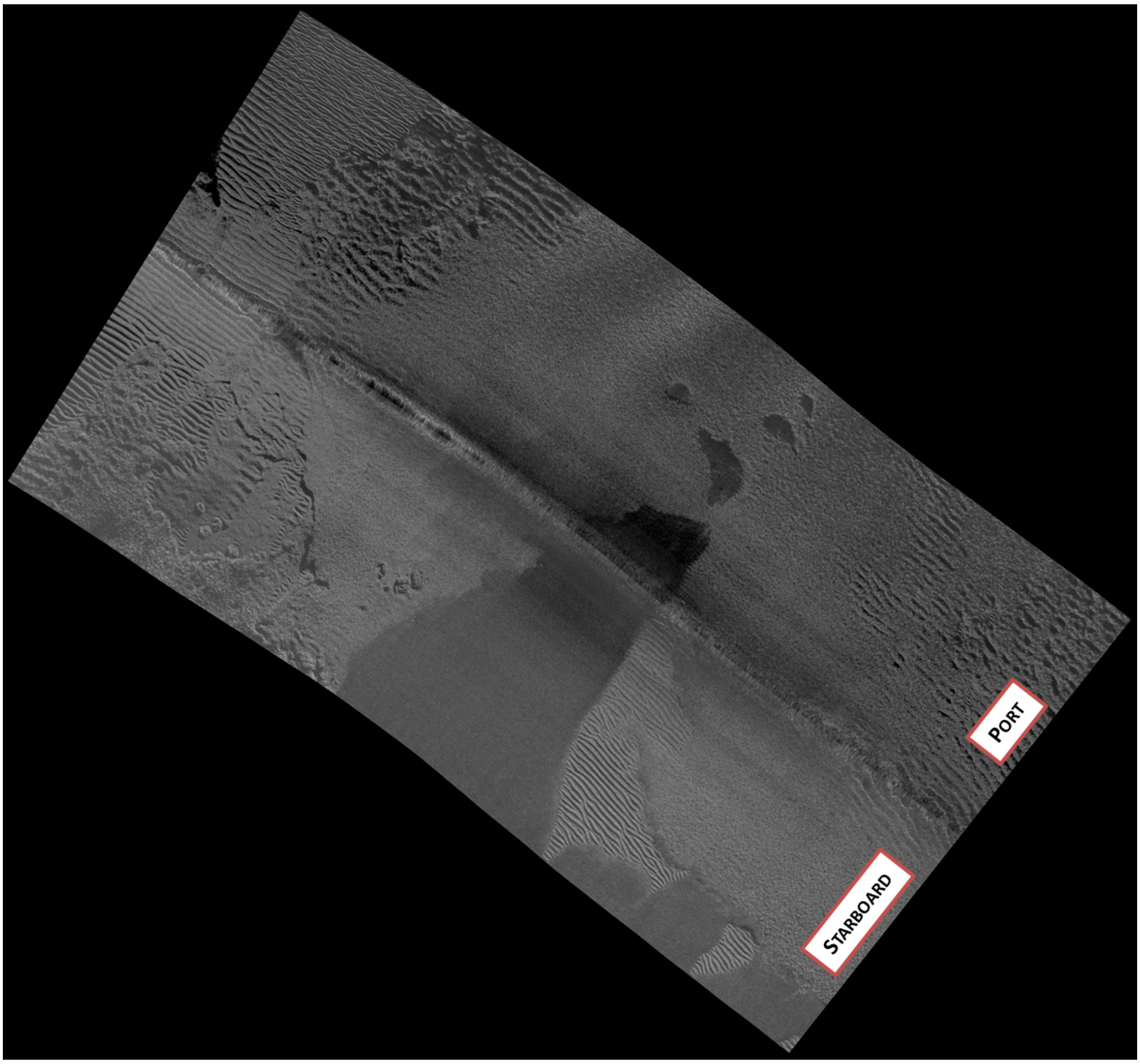


Figure 3.5: Side Scan Sonar Image Generated After Projecting the Acquisitions Over a Georeferenced Raster.

3.2 Image Registration

3.2.1 Symbolic Approaches

While in the past, side scan sonar image registration techniques were used to simply compensate for the slight drift accumulated by the INS, they still represent today a relevant toolbox for addressing the SLAM problem. Historically, two families of methods emerged, the symbolic and the iconic approaches. While the first one is composed of algorithms that extract features to describe the sonar images content and match them, iconic methods rely on the analysis of the intensity distribution of sonar patches and the use of a variety of relevant similarity measures to look for correlations in allegedly overlapping acquisitions. In practice, one needs to use both types of methods, especially in the context of a global registration problem.

At a glance, the underwater environment is made of various types of sediments, rocks, plants and, sometimes, man-made objects. An intuitive approach for performing data association consists in trying to detect either the variation of textures and their respective borders, or salient objects that are big enough to cast a shadow. Detecting, describing and matching such features is the aim of symbolic approaches.

The segmentation of Echo-Shadow complexes has been one of the earliest method for registering side scan sonar acquisitions [27]. Despite the fact that a target's shadow is depends on the grazing angle, it is possible to detect groups of nearby objects that create a unique feature by generating a specific pattern with their combined shadows. These groups of objects can then be matched with overlapping images by detecting a similar pattern to enable registration. One of the challenge related to that approach is the decrease of the SNR over the across track direction. It becomes more and more difficult to differentiate the shadow from the echo as the signal faints. On top of that, for a given size of object, the shadow lengths increases with the distance from the nadir. Consequently, it might be impossible to measure the full extent of the shadow as it is located out of the ping range. Echo-Shadow segmentation is also critically important for applications such as Mine Countermeasure Warfare. In [82], the authors are able to detect underwater mines laying on the seafloor by recognizing automatically the shape of the mine. Other works [65] have shown that it is possible to operate a classification of man-made objects and natural features using the shadows properties. In the study presented in [6], the edges contained in sonar images are detected using a cascade of Haar-features as initially suggested by [102]. This approach is applied to a REMUS AUV in mission for navigation purpose.

While many previous methods were relying on supervised classification algorithms, the work presented in [57] uses a mean-shift clustering model using different properties extracted from the shadows. This unsupervised approach is interesting as it does not require to define a set of prior classes. In fact, while textures may have redundant nature as it is a macroscopic feature, echo-shadows on contrary can be quite unique. Not to mention that some modern underwater mines have varying shapes to avoid detection. More recently, a framework using a Convolutional Neural Network working jointly with a supervised Support Vector Machine classifier was suggested by [108] to improve the efficiency of such segmentation algorithms. In a different field of Remote Sensing, [2] combined the use of Haar features and a CNN for ship detection using Synthetic Aperture Radar data. While the SAR and the SSS are very different sensors, they have a similar geometry of acquisition. This could be an interesting lead to investigate further. It is expected that these type of deep learning techniques will become more and more prevalent in the coming years. Currently one of the challenge in the underwater environment is the lack of massive amount of data for training deep learning algorithms. However, this problem might also be tackled by the recent multiplication of AUVs.

Apart from Echo-Shadow features, underwater textures represent the other prominent semantic features of the seafloor. They make an interesting high-scale image descriptor as well. Like for other types of images different approaches are possible: frequential, pattern-based or on convolutional-based techniques. For instance , [60] applied a bank of Gabor filters to the sonar images and used the outcome as a feature descriptor for classification. This method was initially suggested in the context of classical optical images [42]. In addition to the segmentation problem, the same author used the map of classified texture for registration purposes. She noted that while overlaps on texture classes could give some insights about the overall consistency of the scene, it was not accurate enough to drive the full registration process. Furthermore, some textures, like ample sand dunes could vanish from images due to the grazing angle of the vehicle. This phenomenon introduces a systematic error in the classified map. Unsupervised classification using Local Binary Pattern descriptors and active contour was suggested by [62]. In [22], the authors present another unsupervised classification algorithm using Discrete Wavelegths Transform for producing a multi-resolution representation of the image and performing a K-means clustering over the different classes. Once again, unsupervised approaches might be preferable as they are generating classes themselves, which sounds perhaps more adaptable. For instance that's a limitation of the Gabor filter approach mentioned earlier [60] as well, as the parameterization of the orientation of such filters has to be set by the user prior the segmentation phase. In [68], an approach combining Fourier analysis over three different frequency bands, and the Haralick parameters [40] (computed from Grey Level Co-Occurrence Matrices) has been suggested to facilitated the choice of the direction of the filters in the bank to apply for segmentation. Deep learning techniques have also been suggested recently, however they show a rate of success that is very class-dependant and may fail to correctly represent some types of sediments [14].

An interesting application of the monogenic signal theory initially introduced by [32] to the sonar image processing case is presented by [80]. It consists in applying a Riesz transform to the image to derive a 2D-generalization of the analytical signal. This outcome can then be used to study specific properties of the image like its anisotropy or degree of homogeneity.

3.2.2 Iconic Approaches

Rather than trying to associate images based on matching features, it is possible to operate at the intensity level directly and find correlations between the underlying pixel distributions of two overlapping images. This is the overall concept suggested by iconic methods. Most of these techniques consist in detecting a patch of interest, estimate its distribution and statistics, and find a match candidate in the other image using a similarity measure. Conceptually, iconic methods offer the advantage of being signal-centered: they do not need to make a prior choice, like in the case of the features used in symbolic approaches. They are also able to sometimes leverage some subtle statistical dependencies that could not be captured via symbolic methods. For instance, when an acoustic wave it a target on the seafloor, multiplicative noise results from different secondary acoustic sources interfering. This noise signature depends on the sediment nature and the shape of the object. This is an example of statistical dependency that can be highlighted by iconic algorithms.

In the category of patch matching approaches, [23] the author tested a series of similarity measures commonly used in image processing at the time. He identified that the coupling [24] of the Mutual Information with the Correlation Ratio [87] (MICR) increased the matching rate in his local side scan sonar image registration framework. The detection phase is not applicable to a global registration problem. Indeed, at the time, the Harris Corner detector and the Saliency Map [44] were used to identify regions of interest for patch extraction and matching. Unfortunately such detectors are not efficient as they are not specific enough as discussed later in this chapter.

MI is part of the families of divergence functions that measure the dissimilarity, (or conversely, the similitude) between different distributions of probabilities. Many other similarity measures based on probability law dependencies could be applied like the Kullback-Leibler divergence [51] or the χ^2 test. Other types of statistical dependencies can be applied like the Moravec Correlation [23] or the Zero Normalised Cross-Correlation [69].

The main downside of these approaches is that they are very computationally expensive under a global registration problem.

There are examples in the literature where symbolic and iconic approaches are jointly used to inform each other and improve the overall registration result. For instance, [66] used texture classification information in order to constrain the search for patch matching association using MI to the same type of seafloor only. A similar approach was presented by [79]: texture classification is executed using unsupervised Kohonen maps and Haralick textures parameters.

3.2.3 The global Registration Problem

A global registration problem, such as the relocation of a vehicle after a breakdown, requires to produce data associations over a very large area because of the post-failure uncertainty level. This is why it is critical to step back and have a look at the "great picture". Indeed, the first task is to actually recover the scene topology. Therefore, macroscopic features like textures or echo-shadow seem suitable since they can provide high-scale information and describe the overall structure of the environment. However one cannot assume that a specific feature will be available when required. Indeed, the underwater environment can either be very diverse or completely redundant. Furthermore, while the echoes of seabed target can make very accurate measurements as the exact distance from the sensor to the top point of the feature is equal to the slant range (no bias from the flat bottom hypothesis), it is more complicated to extract precise points using textures segmentation. Or, in order to generate an accurate map, the SLAM algorithm needs precise measurements otherwise the resulting map may be erroneous or the association could be rejected.

On the other hand, iconic methods exceed at finding the exact point where the similarity measure is maximum, thus giving the coordinates of the ideal local registration. However, it comes at the expense of significant computations that are not simply applicable to large and high resolution mosaics by the AUV during the mission. Such approaches require at least a first approximate knowledge of what are the regions of each image that are supposed to overlap to work well.

In fact, it seems that what is needed to address our operational scenario is an approach that takes advantage of both types of methods and use them wisely. It is important to understand where they perform the best, why, and how to guide the data association algorithms of the SLAM framework towards these situations where these algorithms work at their best.

There is also a multi-scale component in this problem. Indeed, it is critical to select tools that operate high scale processing (scene topology recovery) but also ones to further refine the mosaic at the low level once a first map has been generated by the SLAM algorithm. Not only this would enable to perform a local and accurate registration, but this could also flag regions of the mosaic where the overlap post-SLAM turns to be suspicious. Indeed, it is important to keep in mind that sonar images need a trajectory to be generated. If the trajectory greatly changes after the SLAM run, the sonar images will be impacted. Detecting changes in the quality of the overlap may require to revisit earlier data associations in the vicinity (used by the SLAM algorithm) that could turn out to be erroneous.

The multi-scale problem is widely addressed in the field of traditional robotics via the famous keypoint detectors. Since the development of SIFT [63] (Scale Invariant Features) in the early 2000s, more and more algorithms have been further suggested. These tools aim to produce descriptors that capture the scale and the orientation of specific points in the acquisitions. There are extremely useful in visual SLAM and for object tracking. The orientation information is quite helpful as well: it could be used to infer how the sensor has rotated between two successive acquisitions. Unfortunately they are not particularly efficient with sonar images. This is due to several factors: the important and range-increasing SNR, a single channel (compared to 3 in optical images) and also the variation of the across-track and along-track resolutions. In [55], the authors suggest an framework using such keypoints detectors for an AUV generating side scan sonar images and benchmarked [54] the most famous ones: SIFT [63], SURF [12], ORB [89] & BRIEF [20].

However, these are far from being unanimously used by the underwater community: they have a low repeatability, a complex parameterization and are very sensitive to the effects of the slant geometry of observation. Nevertheless, an interesting feature of these algorithms is that they all rely on detection algorithms that create a scale space representation of the images. A series of filtered image is computed using more and more blurring filters. This series of images is then used for detecting points of interest (blobs) that serve as target points for building the keypoint descriptor. Most keypoint detectors use a Linear Scale Space. It consists in applying a series of Gaussian filters with increasing variances. The bigger the variance, the higher the scale of perception of the image. It is a good approximation of the loss of visual details for optical cameras.

Unfortunately, in the case of active sensor, and particularly side scan sonars, such processing makes little sense. Indeed, Gaussian filtering tends to dilute edges by averaging all nearby pixel. However, edges intrinsically represent a significant part of the sonar image information. Blurring such images using a strong variance Gaussian filter may completely erase sand ripples or some echo-shadow complexes for instance. Preserving such information is therefore key for analyzing the whole scene.

An interesting alternative have been suggested through the Non-Linear Scale Space representation that is originally based on the heat diffusion principle [76]. The filters that are applied to the images detect and preserve the edges by reducing the "transfer" at these interfaces. KAZE [3] is an example of keypoint detector that relied on such scale space detector. Such approaches have been applied to Synthetic Aperture Radar images with success for object detection [100]. It is therefore the reason why we decided to investigate if such detector could be used for instance to complement previous iconic registration techniques, like MICR, approaches and make them more applicable to the global registration problem.

3.3 Finding Underwater Orientable Features for the Navigation Algorithm

In order to detect regions of interest to perform data associations at a global level, it is critical to study the behavior of the similarity function that is being used. Since our objective is to make the MICR patch-matching method more efficient and reliable at a high scale level, a series of patches representing characteristic salient features of the environment has been extracted. For example, we considered: interfaces of different features, the echo part of an echo-shadow complex, or salient objects surrounded by a regular texture such as sand ripples.

After this, we computed the MICR score of each query patch over the full mosaic generated by another track. It is important to highlight that the features contained by the query patches lay within the overlap region of both tracks. Consequently, for each patch of the first track, there is a unique corresponding region in the other image. The aim of our experiment is to test if MICR is able to correctly make that association with a high degree of selectivity at the global scale. Indeed, we would like to avoid having a lot of local extrema of the MICR score all over the mosaic. This would mean that there could be too many possible erroneous data associations.

The following figures, 3.6 and 3.7 present the outcome of this experiment for a couple of different query patches. The MICR score is plotted as a heatmap where the yellow color corresponds to local extrema (maximum). In order to assess the degree of selectivity of the MICR similarity measure, we also generated a masked version of the heatmap that retains only values greater than 80% of the maximum score. A single extremum would mean that there is a unique reliable candidate for data association, while a diversity of local extrema would imply different data association hypotheses. Furthermore, the use of different contrast improvement techniques is also benchmarked to assess if the selectivity of MICR is impacted or not.

The results of our experiments could be summarized in two very different trends. First, most of the samples really failed at finding a global maximum score as presented in Figure 3.7. Indeed, complex shapes or periodic textures tend to produce non conclusive heatmaps: a lot of scattered local maxima with low-score in their vicinity.

However, a more promising result is presented in Figure 3.6. Here, the query patch contains a clear edge between two rather homogeneous textures in a flat region of the seafloor. The MICR heatmap presents a unique extremum surrounded by a high-score neighbourhood. It is even more accentuated when edge-preserving filters, like the Non-Local Means filter, are applied to the image. These features are suitable candidates for a global registration problem, therefore it is critical to understand why the match is good with MICR and how to automate their detection for the navigation algorithm.

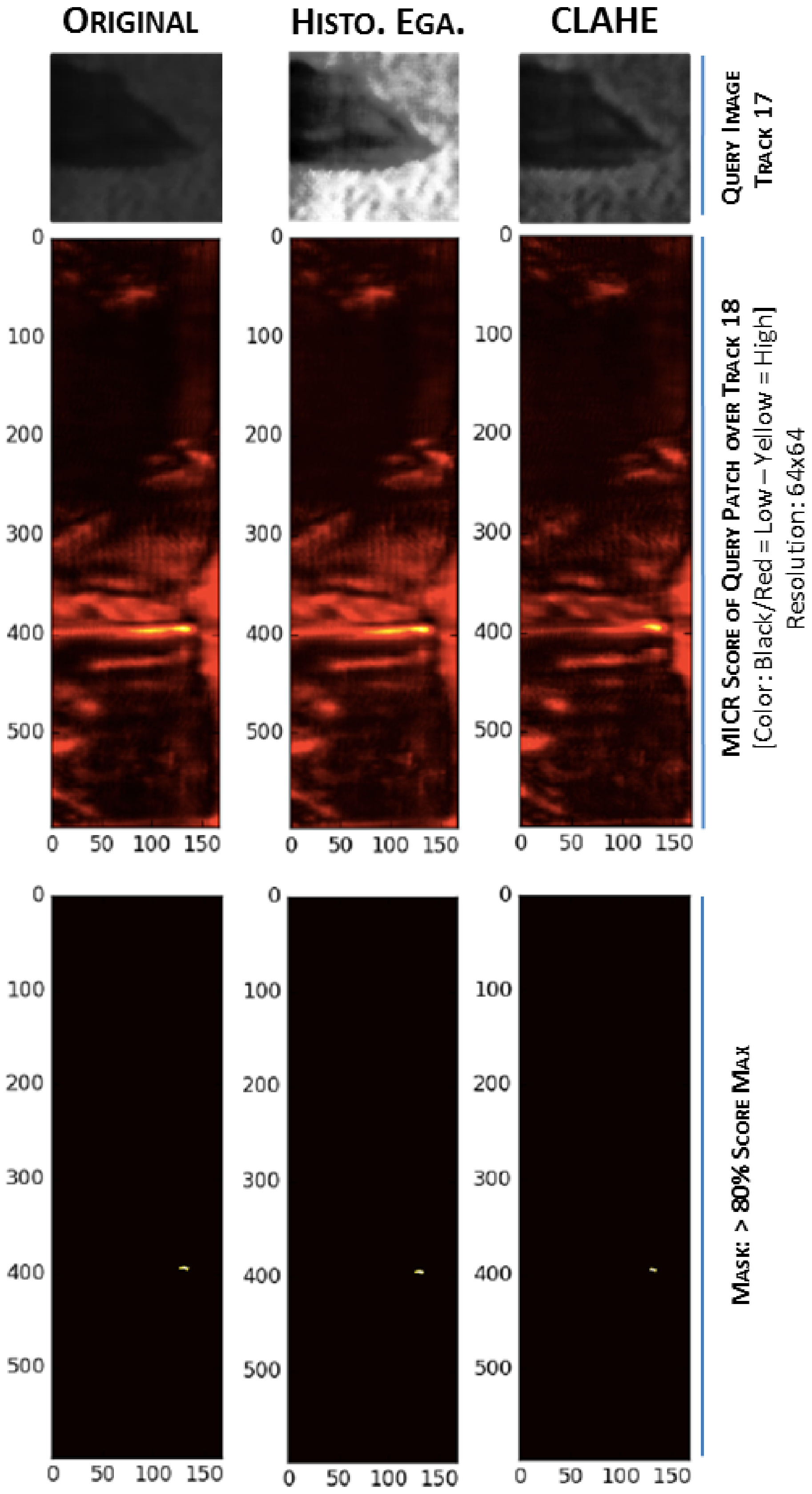


Figure 3.6: The MICR similarity measure performs rather well on this type of patch. The maximum score is indeed matching with its corresponding feature within the other track.

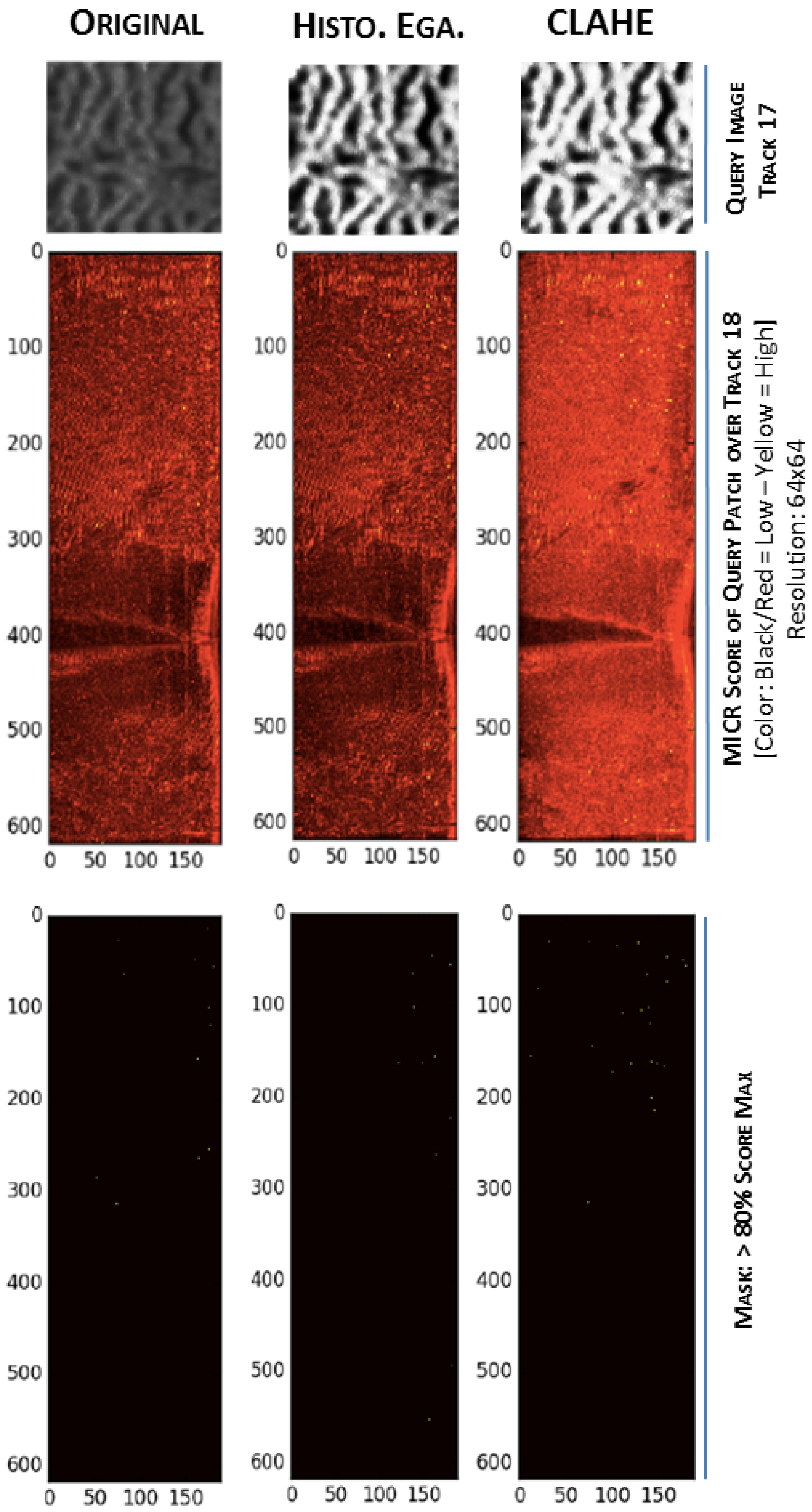


Figure 3.7: In this case, the similarity measure is unable to identify potential matches under a global search over the full track 18. It is quite representative of the outcome of most patches tested.

In order to understand why this similarity measure perform rather well on this type of feature, one can analyze the evolution of the probability mass function (pmf) associated to the patch. In particular, it is critical to capture why the use of different type of edge-preserving filters tend to improve the selectivity of the association process.

Noting that the Mutual Information is part of the MICR similarity measure, the following term is therefore computed to get the similarity score of images X & Y:

$$MI(X,Y) = \sum_{x \in X} \sum_{y \in Y} p_{(X,Y)}(x,y) \log\left(\frac{p_{(X,Y)}(x,y)}{p_X(x)\, p_Y(y)}\right) \tag{3.1}$$

In fact, as highlighted by the previous equation, another key term is the **joint distribution** of the two patches, $p_{(X,Y)}$. Therefore, understanding how this distribution varies with the use of different filters may explain why MICR could produce better results using some prior image processing.

To further investigate this matter, we present in Figure 3.8 the impact of different filtering techniques over the patch that proved to be well performing for data association using MICR. On the left side, the raw sonar image is presented along with its estimated pmf using the histogram technique. The latter looks like mono-modal and pretty much center around a mean value. On the last row, the joint distribution computed with its matching patch on the other track is presented. The joint histogram technique is used to estimate this distribution. At a glance, the joint pmf also appear quite diluted over a large region, therefore explaining a rather low MICR absolute score when no pre-processing is applied to the images.

On the other hand, the removal of the speckle noise by an edge-preserving filter turned the initial pmf into a bimodal function as illustrated on the right part of the figure. Furthermore, It is interesting to note that the post-filtering joint distribution is a lot more concentrated and becomes bimodal as well. This is the reason why the MI score increases with filtering: the more concentrated the joint distribution is, the higher the individual terms of the MI function are.

These observations suggest that some features can become more prominent after the use of edge-preserving filters. Furthermore, they present another interesting advantage: they are relatively flat (no local echoes/shadows pattern like for ripples). Their visual aspect is therefore less dependent on the grazing angle of the sensor. Consequently, they could be seen as helpful in addressing the global registration problem.

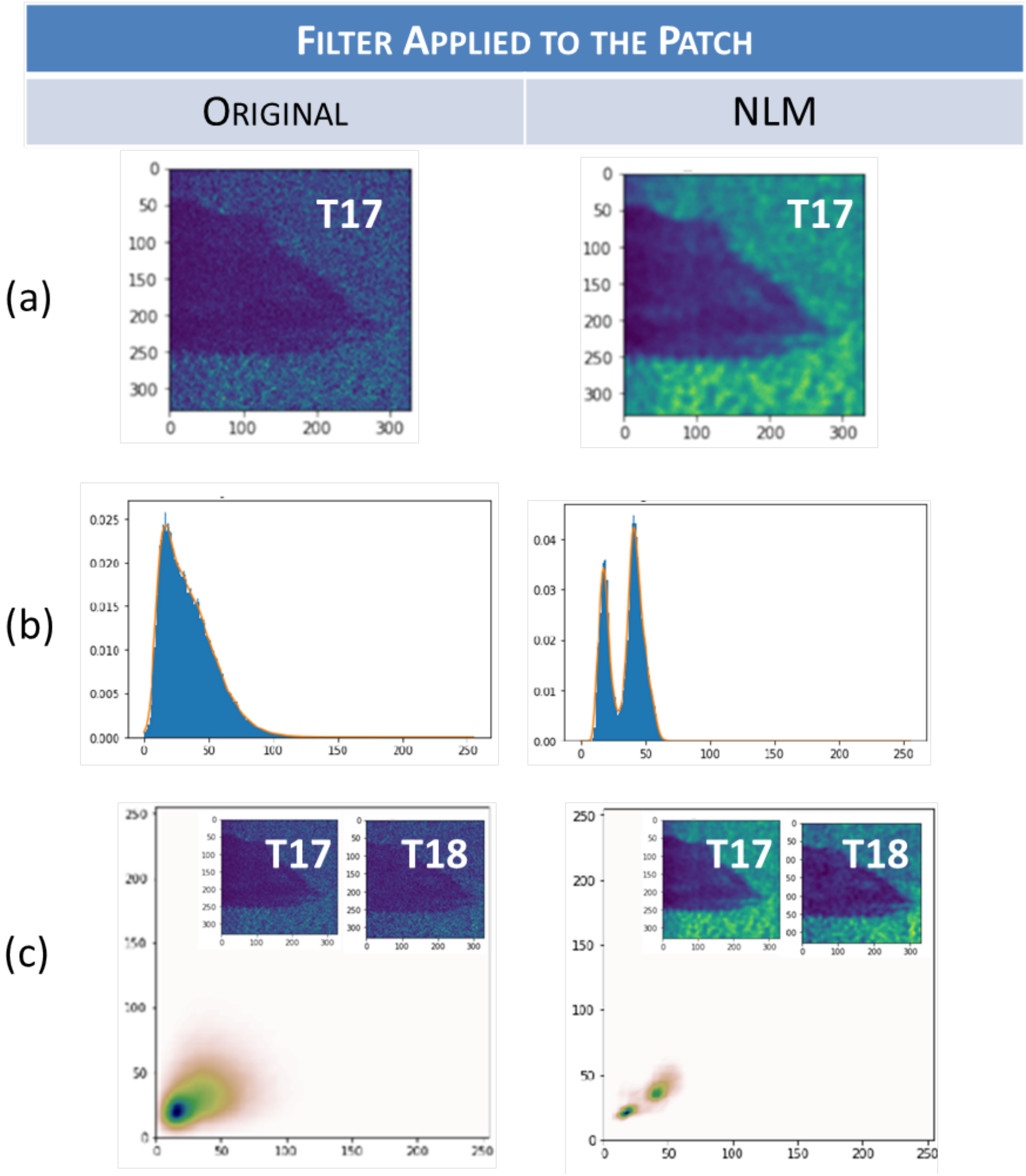


Figure 3.8: Influence of edge-preserving filtering over the probability mass function of sonar patches performing well with MICR, using a NLM filter.

(a): Extracted Patch
(b): Probability mass function estimated via: the Histogram Method (blue) & a Kernel Density Estimator (orange)
(c) Evolution of the Joint Distribution computed using the patches and their counterparts on the other track.

Table 3.1: Filter Parameterization for Feature Detection

FILTER	Parameters		
Gaussian	Window Size	σ_x (Axis x)	σ_y (Axis y)
	(5x5)	75	75
Median	Window Size	-	
	(5x5)		
Bilateral	Neighbourhood Diameter	σ_c (Color)	σ_s (Space)
	9	75	75
Non-Local Mean	h: Strength	Search Window	Patch Size
	30	(21x21)	(7x7)

Since they could be useful points for locating the vehicle using the SLAM algorithm, it is important to investigate if it is possible to detect them under a multi-resolution context. Indeed, being able to detect and select large or small features at will, is critical for searching the correct scene topology. Large and macroscopic features would be for instance preferable in the first place for recovering the overall structure of the environment while smaller ones could be used for finer registration.

In order to address this multi-resolution problem, we used the detector of several keypoint algorithms and measured their efficiency at detecting this kind of structures. It quickly appeared that the outcome of such algorithms was significantly dependant on: the pre-processing of the sonar images, and the keypoint detector parameterization.

We tested several types of filters to study how their combination with classical detectors could improve the feature detection for sonar images. Some of these filters show interesting edge-preserving properties such as the Non-Local Means and the Bilateral ones. The parameters used for our tests are presented in Table 3.1

Contrast improvement is another factor that impact the quality of detection. This is even more critical when considering large area: in this case the decrease of the SNR on the across-track direction becomes non negligible. For our experiments we tried 3 different methods: no change, histogram equalization and the CLAHE approach. The parameters of the later one are described in Table .

Table 3.2: Contrast Improvement Parameterization for Feature Detection

CONTRAST	Parameters	
CLAHE	Clip Limit	Tile Size
	2.0	(8,8)

Table 3.3: Parameterization of the Detectors

DETECTOR	**Parameters**			
Harris	Neighbourhood Size	Aperture	Harris Param.	Selection Criteria
	2	3	$4 * 10^{-2}$	> 0.5 Max
Saliency	Selection Criteria		-	
	> 0.85 Max			
SIFT	Octaves	Contrast Thres.	Edge Thres.	σ
	3	$4 * 10^{-2}$	10	1.6
KAZE	Octaves	Layers	Diffusivity	Threshold
	6	5	Weickert	$2 * 10^{-4}$

The detection tests show an interesting pattern, in particular when edge-preserving filters coupled with contrast improvement techniques are applied to the patches. In Figures 3.11 and 3.12, the SIFT and KAZE detectors were used for identifying local keypoints. Despite the fact that SIFT relies on the linear scale space method whereas KAZE uses a non-linear one, they both managed to capture the same keypoint at the heart of the image structure after using the NLM filter.

In comparison, the Harris Corner detector (Figure 3.9) and the extrema of the Itty & Koch Saliency map (Figure 3.10), as previously used by [24] for sampling patches for local registration using MICR, fail to provide consistent anchor points. They are furthermore too numerous and not discriminating enough for a global registration problem.

It is important to note the intrinsic difficulty of determining an optimal parameterization of the different algorithms. A heuristic approach consists in determining a set of parameters that presents a certain robustness for certain types of features. For example here, we rely on the fact that we face a mostly flat region which is rather well suited to the use of strong filters as long as the global structure of the object is preserved (via its edges and corners). It raises the idea that it is better to operate different specialized detection schemes in parallel on the mosaic, rather than trying to have a too generalist approach.

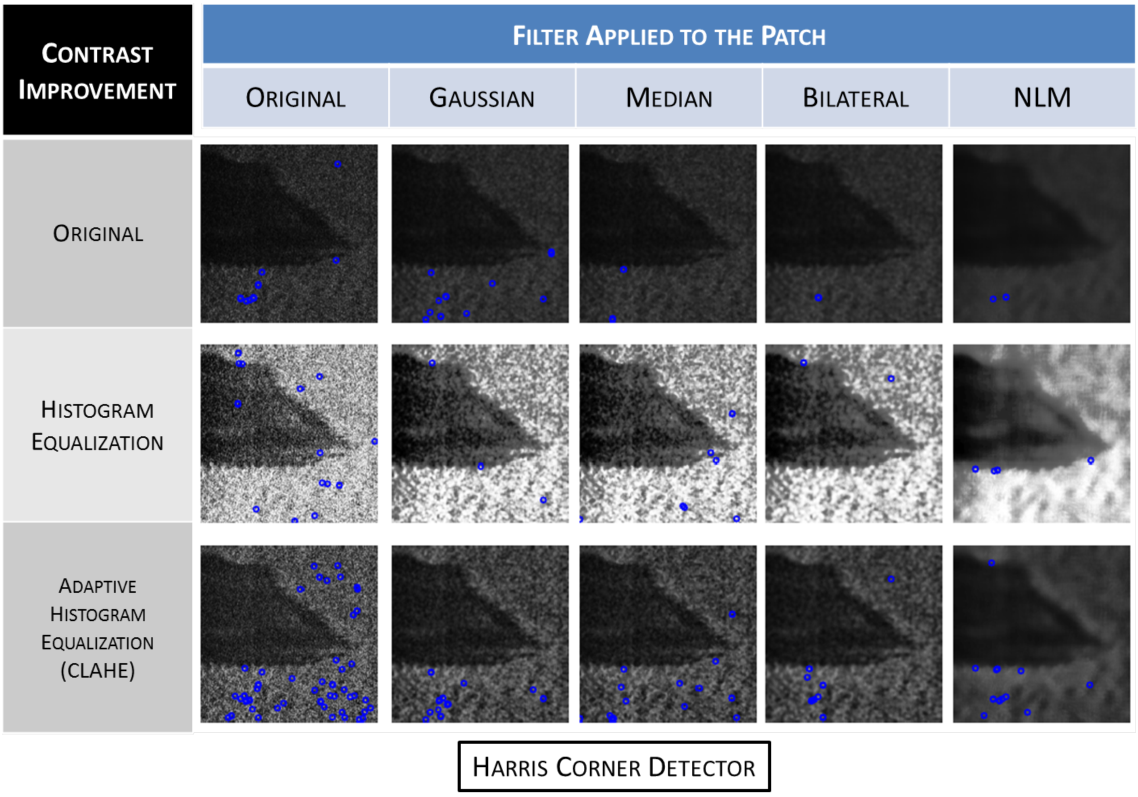


Figure 3.9: Result of the Harris Detector for different processing.

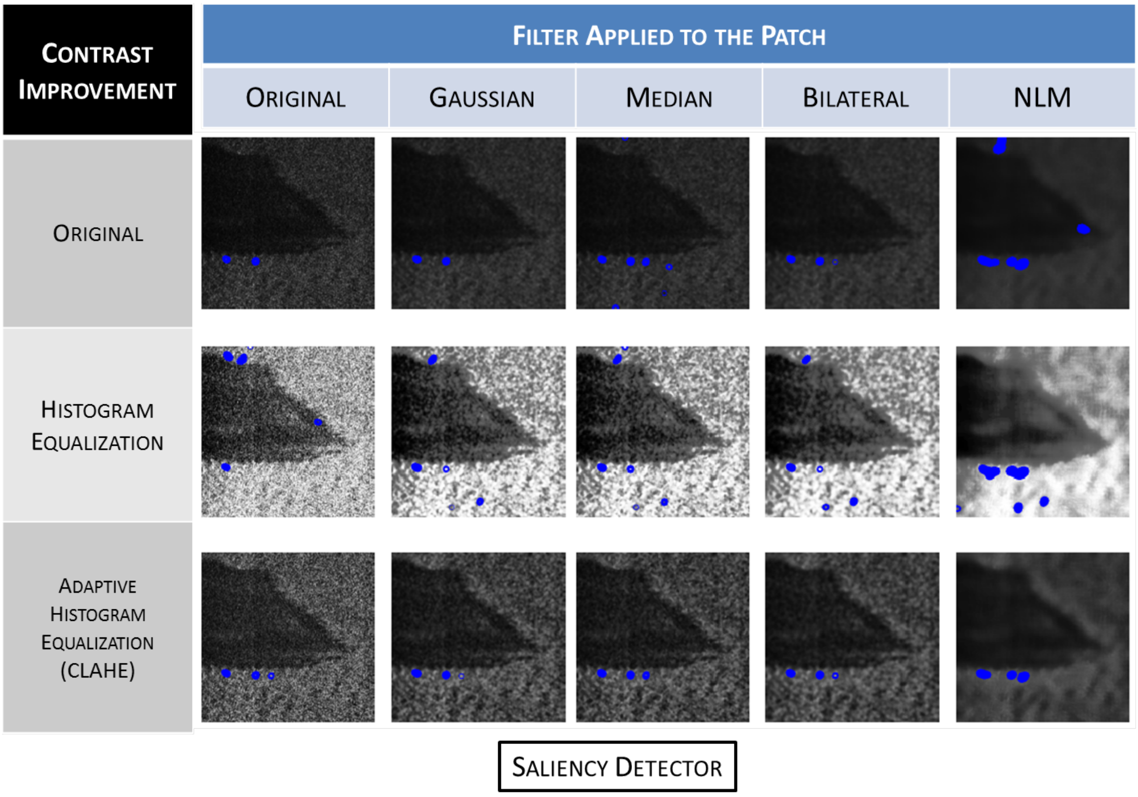


Figure 3.10: Result of the Saliency Extrema for different processing.

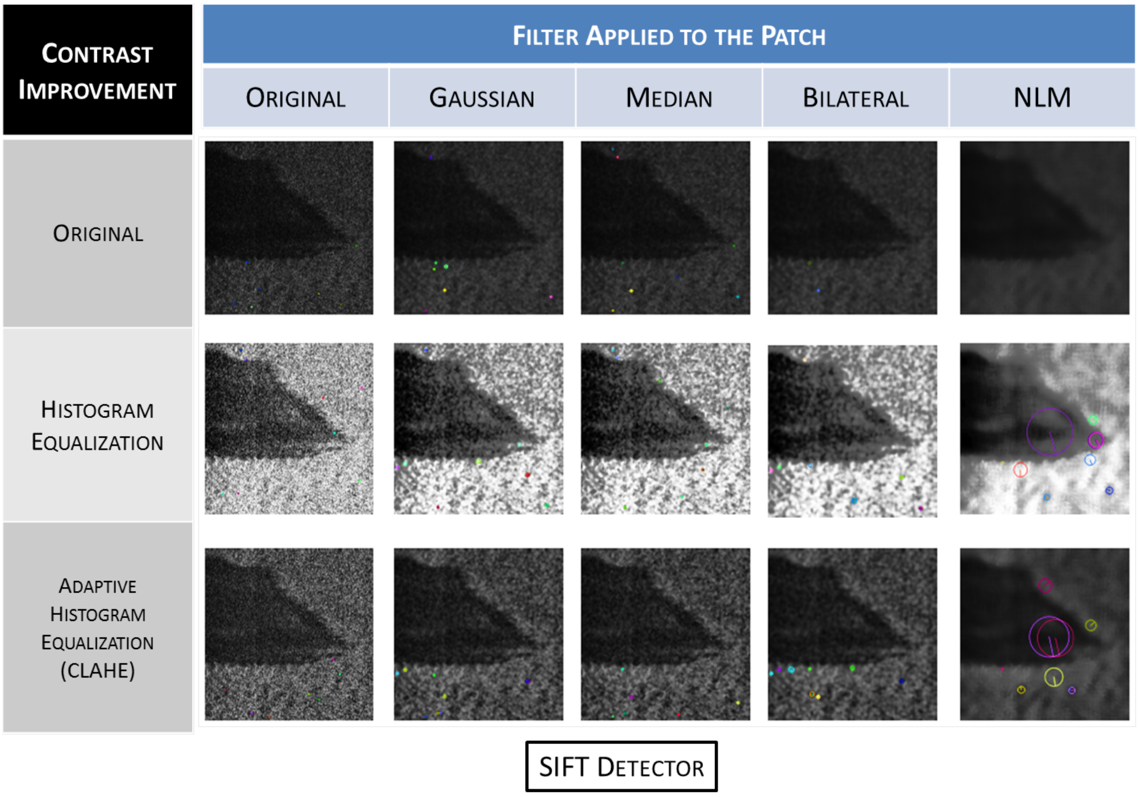


Figure 3.11: Result of the SIFT Detector for different processing.

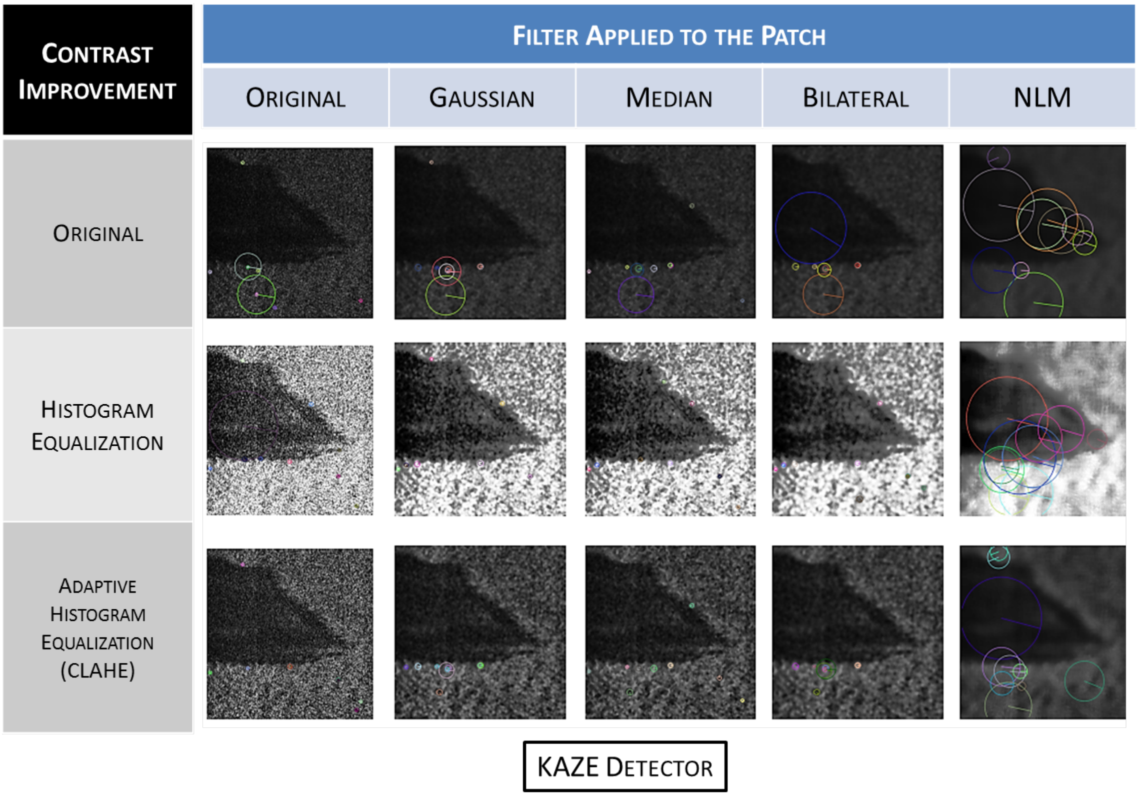


Figure 3.12: Result of the KAZE Detector for different processing.

Another issue arises when processing the full image: such detectors tend to also produce large amount of keypoints in regions where registration is doomed to fail. Indeed, any keypoint located within a regular pattern like sand ripples cannot be used as every point will look similar to each other in the vicinity. This is why, in a similar way to what was described in other studies previously mentioned, we use the textural information to complement our approach.

Consequently, we created a database of patches using a subset of our sonar dataset. In addition to the usual classes selected in the literature for the segmentation of sonar images [83], we also tested classes that were made from the interface of two specific textures. If the keypoint falls into these regions, this would increase the relevance of extracting the patch around directly for instance. Figure 3.13 presents examples of patches that were used during this study. We tested different approaches for the method of patch texture features extraction, the number of classes used and the type of classification algorithm.

Several Track acquisition have been set aside in order to create a training database of textures for segmenting the tracks used in our scenario. Interfaces between different textures were also considered as candidate classes as presented in Figure 3.13. 100 different patches have been extracted per class.

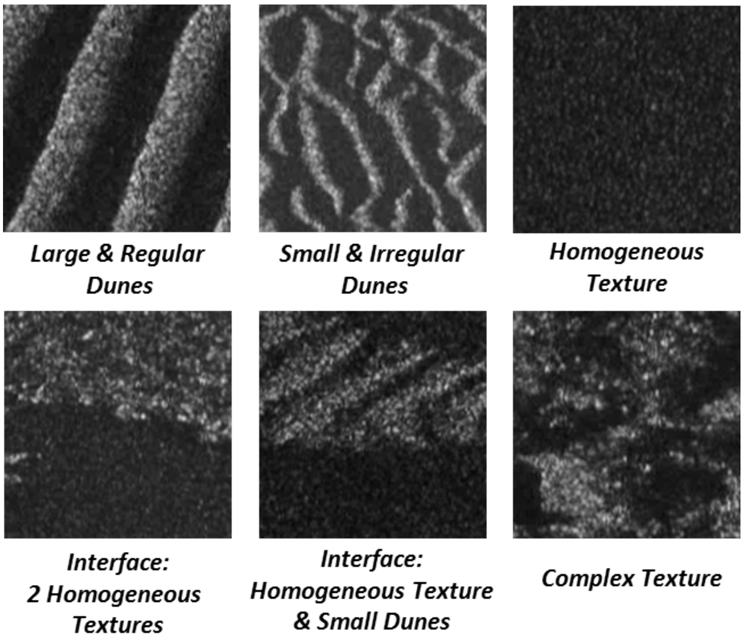


Figure 3.13: Different Classes of Textures Observed in our Dataset.

We also tried to use a Convolutional Neural Network as a texture classifier, unfortunately the results were disappointing. The low level of available data might have played a part on the training accuracy of such method. We made a texture descriptor that was based on Haralick parameters and Local Binary Pattern statistics. The training was executed by a Support Vector Machine (SVM).

Figure 3.14 shows an example where we used 5 different classes: "Homogeneous", "Large Ripples", "Small Ripples", "Interface Homogeneous / Ripples" and "Interface Homogeneous / Homogeneous". In some region this classification performs well as presented in the top detail of the aforementioned Figure. However, it is also far from producing perfect results and more sophisticated approaches should be applied to improve the accuracy. We also noted that using only 2 classes (Homogeneous Vs the rest) was quite efficient at capturing the scene structure.

In Figure 3.14, the top image detail on the right side shows that the segmentation worked well in detecting the border of textures (light green between homogeneous regions (dark purple) and large sand ripples (green/blue). Below, we can see that the variation of dunes amplitude is also captured by the segmentation algorithm.

The texture information could be used in many different ways for improving the understanding of the environment and the texture map could support the navigation process itself. The texture map can also be used as a mask to make other image processing techniques faster and more efficient.

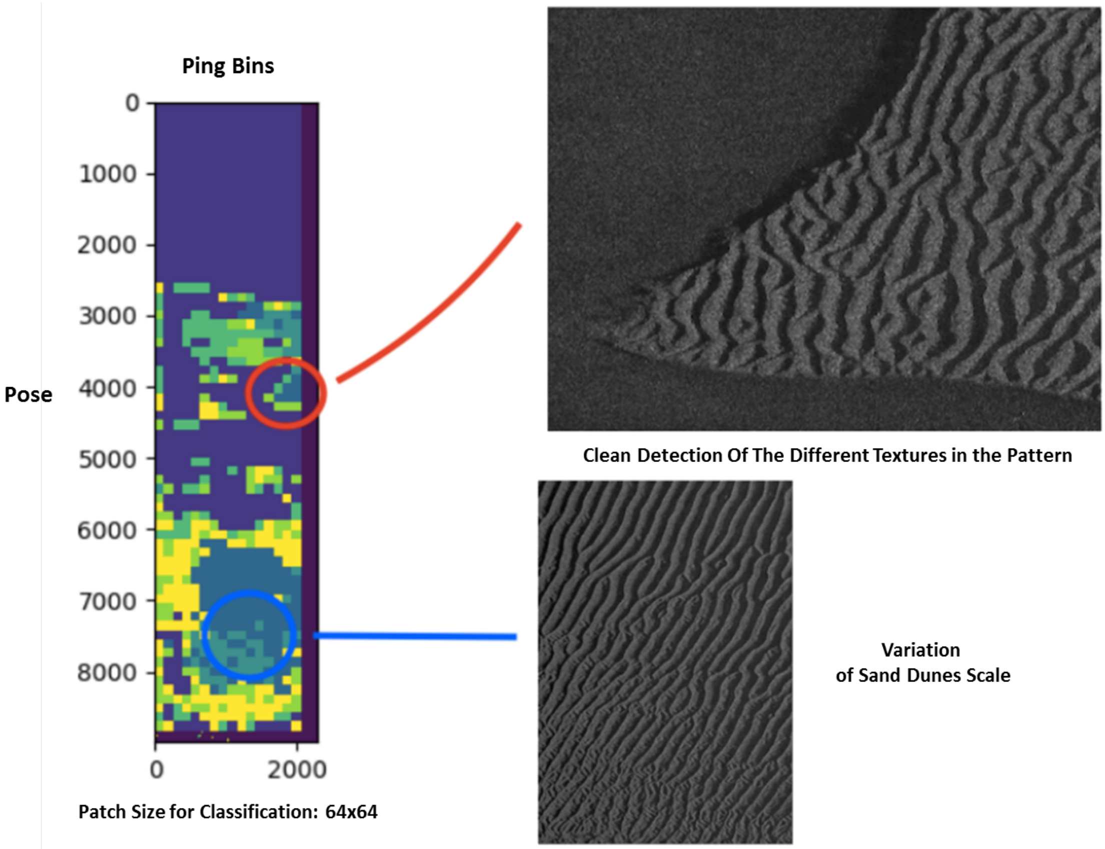


Figure 3.14: Texture Segmentation of Track 17 (Starboard).
The SVM has been trained using textures features derived from
Haralick Parameters and Local Binary Patterns, assuming 5 classes.

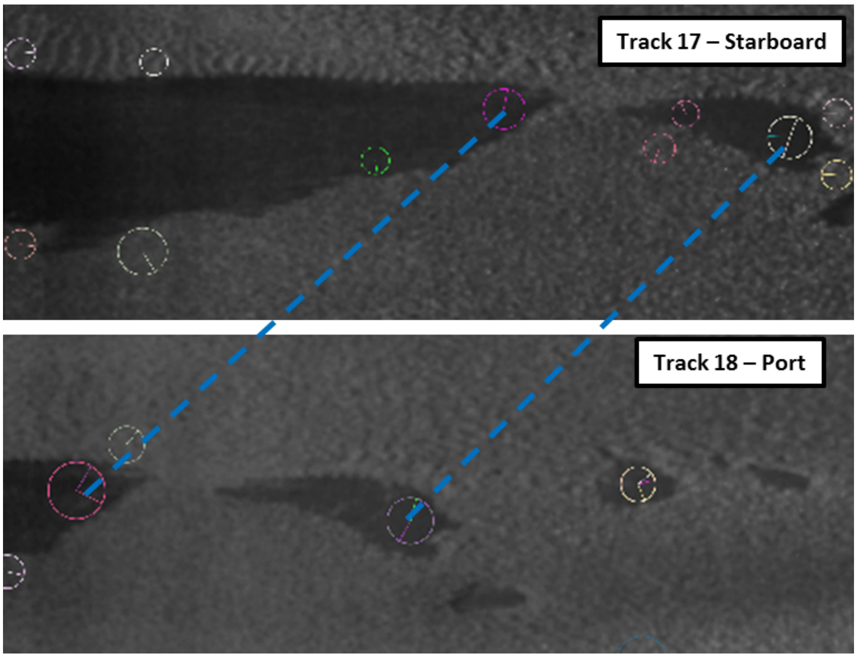


Figure 3.15: Result of SIFT Detector over Track 17 & Track 18 after using NLM filter, Contrast improvement (CLAHE), and Textural information.

When coupling the search of keypoints in the vicinity of flat textures regions after applying edge-preserving filters, we end up with the result presented in Figure 3.15. We can see that we managed to produce redundant keypoints detections on two overlapping images. These experiments show the journey from trying to overcome some limitations of the similarity measure and the patch extractor algorithm by understanding why they can perform well or not.

This is useful information for constraining the use of some algorithms is specific regions where they operate well. Furthermore, it may sound reasonable to try to go further and develop a family of keypoint extractor that are more appropriate for active sensors such as the side scan sonar. In addition to that, it is interesting to observe how the NLM filter improves the quality of detection and matching. Possibly a non-linear scale space representation of sonar images could be derived from it.

3.4 Discussions & Conclusion

Overall, performing data associations in the underwater environment is a challenging task, especially under a global registration problem. While the echo-shadow complexes can make very discriminating points they could also be rare. Furthermore, the variability of the features due to the grazing angle increases the difficulty. Some textures, like sand ripples, are highly sensitive to that parameter and can vanish from an image to another. Therefore, the structural data of the image is subject to variation and poses a threat for the AUV that needs to find its location after a potential breakdown.

By analyzing the performance of traditional iconic methods, we identified regions in the sonar images presents a lower dependency to the change of the sensor location and tend to react well to iconic patch-matching methods after applying edge-preserving filters and contrast improvement algorithms. Relatively flat and homogeneous, they look like an interesting complement to echo-shadow points for finding global features to recover the scene topology.

Performing an efficient detection is also a critical point. In order to perform a large-scale screening of the image, we investigated some techniques used by traditional keypoint extractors that rely on linear & non-linear scale space detectors. By applying edge-preserving techniques and textural information, it is possible to limit the detection to a lower number of points in space that can be further analyzed using traditional patch-matching techniques.

However, this type of approach cannot solve on its own the global registration problem. It is in fact a complementary technique that should be used jointly with other large-scale data association algorithm like echo-shadows detectors. The ideal framework should use a variety of complementary techniques that are applied in regions where they are supposed to perform better. Having an association method that is using elevated features (like echo-shadow complexes) working in parallel with another one focusing on rather flat region, using patch-matching, could be an interesting strategy.

Furthermore, we can infer that the SLAM algorithm will also play a decisive part. Indeed, based on these observations, data association in this context is prone to error. It is therefore critical to address the problem of outliers generated by the data association phase within the SLAM module as discussed in the following Chapter and implement some safeguards against these outliers.

Chapter 4
Underwater SLAM

Highlights of the Chapter

- Navigation consists in producing an accurate estimate of the vehicle's position in relation to a reference map. This map can be depicted in various forms. In uncharted waters, no prior representation of the environment is available. As a result, the autonomous system has to rely on its perception of its surroundings using the different on board observation instruments. These data can be processed and used to create a map that could be later used to relocate the system, when needed. While there are different approaches to create a map, it is highly dependent on the type of remote sensing tool used.

- The mapping process is a complex problem as the interpretation of each observation depends on the actual location of the system. Uncertainty on the vehicle's position directly impacts the process of creating the map. Indeed, neither the true location of the map's landmarks or the vehicle is known at any anytime. Consequently, there is a joint estimation problem where both the map and the vehicle locations are computed. This problem has been widely addressed during the past 25 years under the name of Simultaneous Localization and Mapping (SLAM).

- The first implementations of the SLAM algorithms were derived from the different data fusion algorithms used for state estimation. As introduced in the Chapter about Underwater Navigation, many Bayesian filtering techniques like the Kalman or the Particle Filters were extended to the SLAM problem. For instance, in the case of the EKF, the state vector could simply include the location estimate of different landmarks that are used to describe the map. SLAM presents many challenges as it is a very operational driven field of research. In particular, data association plays a critical part for SLAM and may be the source of inconsistent maps if not performed successfully. Unfortunately, on top of already being one of the most complex challenges of robotics, SLAM is highly vulnerable to poorly structured environments, such as the underwater ones.

- In conjunction with the uprising of field robotics, new SLAM strategies emerged in the 2000s. In particular, graph-based approaches thrived as they are exploiting the sparse nature of the SLAM problem. Indeed, the dependencies between the different poses of the vehicle are generally limited to a couple of motion constraints between successive poses and some occasional observation constraints. Graph-based approaches consist in translating these constraints into the form of edges linked to nodes representing the system poses. Rather than trying to describe the probability density function of the poses and the map as it is done in Bayesian filtering for state estimation, graph SLAM is defined as an optimization problem where the objective is to find the set of poses that verifies the *Maximum A Posteriori* of the SLAM probability function. It generally involves solving a weighted non-linear least square problem and handling sparse matrices (due to the low number of interconnections between poses). Several iterative optimization techniques take advantage of these specific properties thus allowing to address very large scale SLAM problems.

- As this modelling became more and more predominant in the SLAM community, significant research efforts were made in the 2010s to robustify graph SLAM techniques especially against outliers resulting from erroneous data associations. Historically, SLAM frameworks have been split into two parts, the **front-end** that deals with data fusion and association and the **back-end** that focuses on the optimisation problem. The front-end is very sensor-dependant as data associations strategies strongly rely on the type of sensor used. Unfortunately, it is impossible to guarantee that no false positives are forwarded to the back-end. Any erroneous loop-closure constraint leads to an inconsistent graph topology that ultimately produces incorrect map and trajectory estimates. Consequently, it became imperative to explore how to address this issue at the back-end level. The different robustification strategies that have been developed occupy a central place in this thesis as we are dealing with an operational scenario that is highly prone to erroneous data associations.

- Indeed, not only the underwater environment is poorly structured, but the chances of observing the same point several times are low due to the side-looking nature of the active sonar used for mapping the seafloor. The low repetability of observations of natural landmarks dramatically impacts the efficiency of SLAM algorithms, in particular the ones based on Bayesian filtering. Furthermore, as the signature of a seafloor object significantly changes with the grazing angle and the point of observation, finding global observation constraints (or loop-closure) is highly sensitive. This is aggravated by the fact that two matching observations could be made after a significant time interval, which may correspond to an important increase of the navigation uncertainty due to the drift of the INS. Finally, when facing an extreme operational scenario such as a momentary failure, we end up in the *stolen robot* configuration but applied to the underwater environment. The combination of both makes it a dire challenge for the AUV.

- After presenting the key ideas related to robust graph slam strategies, the last part of this Chapter presents several experiments based on a real dataset. The aim is to assess how the algorithms behave in the context of our operational scenario and also explore their limits. After presenting how the graph SLAM modelling works on the ground truth, we study how introducing different erroneous data associations alter the outcome of the optimization phase. We discuss how some of the robust SLAM algorithms are very efficient at removing erroneous data associations under certain conditions. We discuss how the effects of the possible interruption of the motion signal makes the search for an initial scene topology estimate critical to make sure these strategies could work. The conclusion of this section is crucial for defining the specifics of the framework that has been developed during our research. In particular, we conclude that it is necessary to undertake a **scene validation** step to reconsider some potentially erroneous association hypotheses that have predominated during the supposedly robust optimization phase.

4.1 Principle of SLAM and its application in the underwater environment

4.1.1 General SLAM definition

As presented in the chapter about underwater navigation, SLAM aims to solve the problem of having an autonomous system operating in an uncharted environment. As it cannot rely on a previously existing reference map, it has to use its perception of the environment to create its own internal representation of its surroundings. The AUV then refers to this newly created internal map to locate itself during the execution of the mission.

In essence, SLAM is a highly complex problem as it involves many issues. Firstly, it is dependent on the means at the disposal of the autonomous system, whether they are internal, to measure its dynamical parameters (INS), or external, to be able to perceive what surrounds it. The nature of the family of sensors that is being used strongly condition its analysis of the environment. Consequently, the type of map that is generated is very sensor-dependent.

The most important point in SLAM is the interdependency between the estimation of the map and the trajectory of the autonomous system. From a probabilistic point of view, it can be seen as a joint estimation problem as presented below. For instance, if we assume that the map is made of a set of landmarks spotted by the vehicle sensors, the absolute location of these points is unknown. In fact, the robot can only estimate their location when it encounters one of them using its observation model (as described in Chapter 2). However the landmark position is conditioned by the knowledge of the true position of the vehicle which is another unknown parameter that can only be estimated as well…

Unsurprisingly, there are several approaches to tackle this equation. Since SLAM is an extension of the navigation problem described in Chapter 2, there is a clear filiation between the estimation methods used for addressing these two topics. However, since the early 2000s, two distinct families of algorithms have clearly emerged [29] , [10]. In particular, they differ in the mathematical problem they seek to solve. The first one is composed of filtering methods which evolve from the algorithms presented in Chapter 2. For instance, the Augmented State EKF is another derivative of the Kalman Filter and models the map by adding landmarks to the state vector (therefore the name). On the particle filtering side, we can cite the example of FastSLAM [67] which relies on the principle of Rao-Blackwellization to avoid having to manage a too large sampling space by cutting the problem in two: an analytical part (for the map) and a purely particle filtering one (for the trajectory).

Probabilistic Formulation of the SLAM Problem

The general probabilistic definition of the SLAM problem consists in estimating the following probability distribution:

$$\forall t \in [\![0..T]\!] \quad P(x_t, M | U_{0:T}, Z_{0:T}, x_0) \tag{4.1}$$

Where:

- x_t : The vehicle poses at time t
- M : The map (that can be represented by a set of landmarks $L_{[1:m]}$ for instance)
- $U_{0:T}$: The set of the control commands (or motion measurements depending of the convention) between $t = 0$ and $t = T$
- $Z_{0:T}$: The set of the measurement vectors between $t = 0$ and $t = T$

The second main category includes smoothing algorithms, whose objective is to process all the acquired data at the same time in order to compute the map and the trajectory. It is generally a question of solving an optimization problem that often takes the form of deriving *a maximum a posteriori*. Because of the large amount of data to be processed, these approaches are highly dependent on advances in mathematical optimization. This family of algorithms experienced a remarkable boom towards the end of the 2000s through the use of graph modeling. In 2006, the GraphSLAM algorithm [98] demonstrates the potential of this approach over traditional filtering methods like the Augmented State EKF.

Namely, the data association phase is a lot more efficient. In this implementation, the authors combine the search of correspondence with the estimation process of the map, while being able to undo easily any association if deemed appropriate. Graph-based SLAM became more and more popular in the following years. Many following research works focused on robustifying this approach against outliers or making it more time-efficient. An interesting aspect of this philosophy is that the search of observation associations slightly evolved towards the problem of finding a correct graph topology.

Filtering vs Smoothing

The estimation process consists in finding:

In the case of the **Filtering Approach**:

$$\forall t \in [\![0..T]\!] \quad P(x_t, M | U_{t-1}, Z_t) \tag{4.2}$$

Where:

- x_t : The vehicle poses at time t
- M : The map
- U_{t-1} : Last control commands
- Z_{t-1} : Last set of measurements

In the case of the **Smoothing Approach**:

$$P(X_{0:T}, M | U_{0:T}, Z_{0:T}) \tag{4.3}$$

Where:

- $X_{0:T}$ M : The full set of poses (trajectory) & the map
- $U_{0:T}$, $Z_{0:T}$: The complete sequence of Control Commands & Measurements

4.1.2 Use of SLAM algorithms in the underwater environment

First applications of SLAM algorithms in the underwater environment could be traced back to the early 2000's. In 2001, a multi-hypothesis tracking filter coupled with the processing of a sonar dataset has been used by [91]. Later on, smoothing methods were considered for registering large side scan sonar images: in 2004, Ruiz [90], applied a Rauch-Tung-Striebel smoothing filter to a trajectory initially estimated via an EKF. Landmarks were manually associated. This approach was beneficial for merging large acquisitions of side scan sonar and segmenting the different classes of textures [83].

Augmented State EKF is also a popular option in the world of underwater robotics. In [85], Ribas uses this type of algorithm with a sonar image dataset in a structured underwater environment. Such approach was also applied to AUVs equipped with optical down-looking cameras by [30]. Here, the authors detects overlapping images and try to register them under a local registration approach.

Particle filter are proved to be efficient for addressing featureless underwater SLAM [11]. This is the case for AUV equipped with multi-beam echo-sounder deployed for bathymetric surveys. In that case, the acquisitions are represented as a point cloud and the map is not described via singular landmarks.

Graph-based modelling have also been used in that context since the late 2000s/early 2010s. Pfingsthorn [78] conducted outdoor testing using a bottom-looking optical camera and modelled the map under a graph representation. In [13], the authors applied the iSAM framework [48] to the problem of registering large side scan sonar acquisitions using manually extracted associated features while applying a multi-session approach.

Another interesting development of underwater SLAM is its application to the cooperative navigation problem. Indeed, if we assume that a group of AUVs is operating in the same vicinity, they can exchange information in order to refine their own estimate of the map. However, the general approach, as described in [88], does not take into account the underwater communication problem. Indeed, the ocean is a poor communication channel with a narrow bandwidth. Later work studied how this constraint impacts the exchange of information between AUVs [99], [53] for collaborative navigation (CN) purposes. One interesting application of this technique is that it is possible to detect inconsistent data associations and remove them from the current map assumptions. Such concept is being discussed in [5] under the constraint of limited communications.

Overall, these previous works suggest that while there has been a notable improvement in performance in terrestrial robotics, the underwater case remains much more complex. It is only with the rise of underwater robotics in the early 2010s that AUVs can now embark sufficient computing capacity to handle the tremendous flow of data associated with SLAM. Such processes are also highly dependent on a good knowledge of trajectories thanks to the use of accurate INS or ancillary localization systems such as LBLs. In addition, the search for non-local data associations remains a major challenge, especially when the AUV is not navigating near artificial structures that can be used as reference points. In such cases, operational failure remains a major concern since the uncertainty about the position will lead to erroneous associations that may go unnoticed and eventually corrupt the map.

In previous studies, inertial data give somehow access to an almost correct topology of the scene. This backbone plays a major structuring role and prevents the AUV to have to deal with a large number of outliers. The operational case we are addressing is therefore, first of all, a search for the correct topology, allowing to later reduce the problem to a simpler task. This is one of the reasons why a graph approach seems relevant. As it is possible to easily reconsider data associations by removing or adding edges, we imagine that this search for the correct scene topology is facilitated by the optimization of a graph with variable parameters.

4.2 Focus on Robust Graph SLAM

4.2.1 Principle of Graph SLAM

The principle of graph-based SLAM algorithms is rather simple. The idea is to consider that all the poses and reference points on the map represent the vertices of a graph. These different elements are interconnected by edges that model their dependency relationship. For example, two vertices of consecutive vehicle poses will be linked by an edge, or constraint, which reflects the transition from one state to another. This relationship is characterized by the *motion model* as defined previously in the chapter on navigation. When an edge connects two non-consecutive poses, there is a spatial dependency that is identified as a loop closure constraint (LCC). LCCs are very often the result of matching observations of the same point in space by the vehicle at two different moments. This revisiting of a place allows the trajectory to loop back on itself which has a major impact on the topology of the scene.

In the case where the revisited point takes the form of a landmark that is clearly defined in the graph, the dependency between the pose vertex and the landmark is described using the *observation model* as previously mentioned in Chapter 2. In this scenario, two non-successive poses are connected through the landmark which represents a group of 3 vertices for 2 edges (defined by the observation model). This is described in Figure 4.1.

There is another form of modeling that does not explicitly introduce landmarks. In this case only vertices representing the vehicle poses are encoded. This representation is called a **Pose Graph**. Here the LCCs are directly encoded from one pose to another. The mathematical function that defines a Pose Graph is slightly more compact because the observation model does not appear explicitly either. In fact the LCCs are determined in the form of a pseudo-displacement constraint with a parameterization that is derived from the observations. Basically, if the same point is observed at two different opportunities, it is possible to deduce a passage relation between both vertices using the motion model. Important note: the parameterization of such LCC assumes the implicit use of an observation model: however, this is handled upstream at the front-end level, not in the graph optimization phase. This case is represented in Figure 4.2.

Once the graph is constructed by adding vertices and constraints, state variables that describe these vertices are then estimated. Indeed, just like in any other type of SLAM algorithm, the "true" values of the robot positions or landmarks are unknown. It is a set of latent variables. One of the most import characteristics of the graph in this context is that it has a sparse structure. Indeed, for each pose there are only two motion constraints (one before and one after) and sometimes some LCCs. Generally, the number of LCCs is of a much lower magnitude compared to the total number of poses. Consequently, the matrix that maps the dependencies between every poses is highly sparse. This is a critical property for solving the estimation problem as explained in the next section.

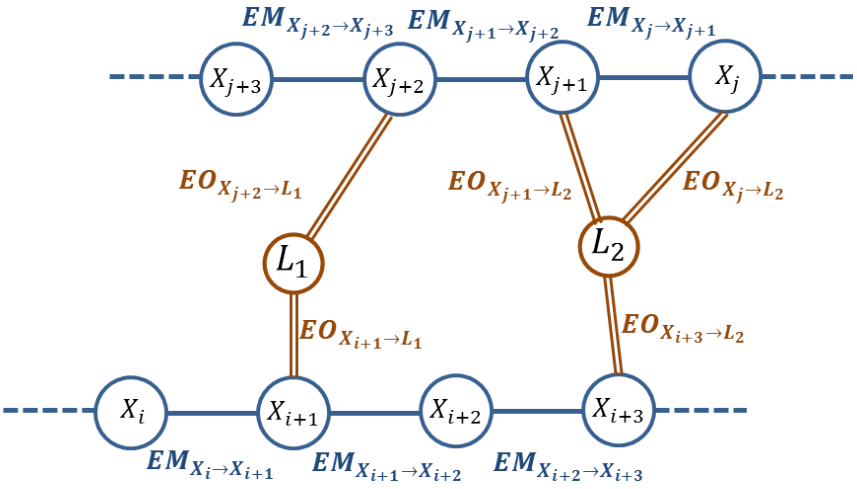


Figure 4.1: Graph modelling using both vehicle poses ($X_{i,j}$) and landmarks (L_1, L_2) as nodes. Each node is connected by one or more edges that represent a constraint. $EM_{X_i\to X_{i+1}}$ represent the motion constraint between 2 successive poses. This edge is related to the motion model of the vehicle and the measurements of the motion sensors (the INS and others). $EO_{X_{i+1}\to L_1}$ is an edge representing the observation of the landmark L_1 by the vehicle at the pose X_{i+1}. With this representation, the map is generally defined by the set of landmarks used in the graph.

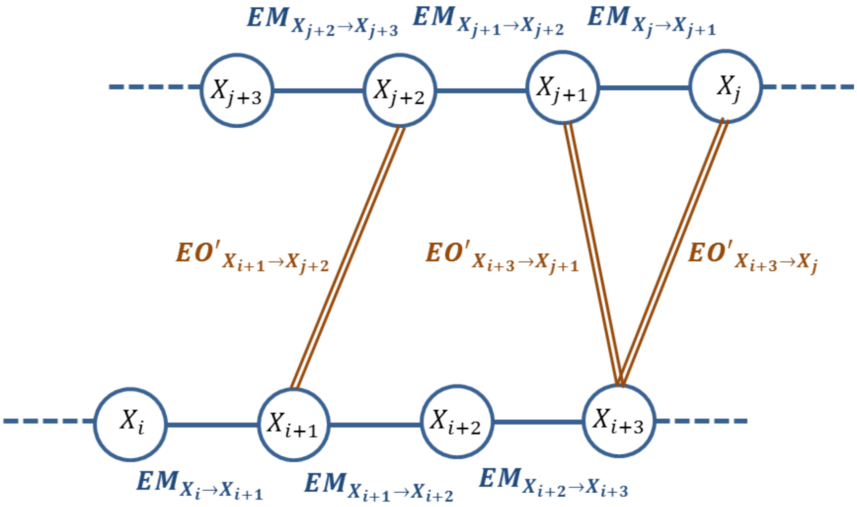

$EM_{X_{j+2}\to X_{j+3}}$
$EM_{X_{j+1}\to X_{j+2}}$
$EM_{X_j\to X_{j+1}}$
X_{j+3}
X_{j+2}
X_{j+1}
X_j
$EO'_{X_{i+1}\to X_{j+2}}$
$EO'_{X_{i+3}\to X_{j+1}}$
$EO'_{X_{i+3}\to X_j}$
X_i
X_{i+1}
X_{i+2}
X_{i+3}
$EM_{X_i\to X_{i+1}}$
$EM_{X_{i+1}\to X_{i+2}}$
$EM_{X_{i+2}\to X_{i+3}}$

Figure 4.2: Alternatively, the observation constraints can be directly encoded between the vehicle poses, thus making the landmarks disappear from the graph structure. This type of representation is commonly known as a *Pose Graph*. These constraints directly encode the spatial dependence between the two poses. When the two poses are non consecutive, these constraints are called *Loop Closure Constraints*. From an operational point of view, they can be encoded as a pseudo-movement from one pose to another.

4.2.2 Estimating the state variables by solving an optimization problem

Once the graph has been constructed, the values of the state variables contained in the vertices must be determined. To achieve this, it is necessary to go back to the probabilistic formulation of SLAM in the context of smoothing. By using Bayes' theorem, it is possible to reduce the SLAM problem to the search of an optimum. This optimum represents the set of poses and landmarks that minimize two error terms: one that is related to displacement constraints and another that relates to landmark and other LCC observations. This approach is described in the equation box on the following page.

When f (the motion model) and h (the observation model) are linear functions, it is possible to derive the optimum directly by using the normal equation. This is mainly a matter of being able to invert the matrix that describes the linearity relationship between the state variables and the model. Due to the complexity of this inversion step, methods such as the QR matrix decomposition or Choleski's decomposition are often used.

In reality, these functions are not linear. Therefore, there is no analytical solution to the problem described above. Consequently, it is necessary to use iterative approximation methods such as the Gauss-Newton (GN), Levenberg-Marquardt (LM) [61] or dog-leg algorithms. The first one uses Taylor's expansion of the optimized function. The aim is to find in the vicinity of the function's expansion what is the variation of (X, M) that locally minimizes the function. The second method alternatively uses the traditional gradient descent and the GN method. Finally, Powell's dog-leg method [81] proposes a technique similar to LM. However, it constrains the search for the new iteration term in a vicinity of the initial point that is considered to be reliable. In fact, where the gradient always guarantees convergence towards the minimum of a convex function, the number of iterations may be much too large, hence the motivation to use other algorithms.

The graph-based SLAM equation

As it is a **smoothing approach**, the following equation defines our estimation problem:

$$P(X_{0:T}, M | U_{0:T}, Z_{0:T})$$

Baye's Theorem can be recursively applied to this term:

$$P(\dots) \propto \prod_t \left[P(x_t | x_{t-1}, u_t) \prod_s P(z_{t,s} | x_t, m_{c_{t,s}}) \right] \tag{4.4}$$

Where:

- $m_{c_{t,s}}$ represents the s-th correspondence made at pose t with the map, which involves measurement $z_{t,s}$

The objective is to find the set $(X^*_{0:T}, M^*)$ that maximizes this probability distribution. This is the **Maximum A Posteriori**. This set is also the value for which the negative log of the above expression is also minimal:

$$(X^*_{0:T}, M^*) = \underset{X_{0:T},M}{\operatorname{argmin}} \sum_t \left[-\log \left[P(x_t | x_{t-1}, u_t) \right] + \sum_s -\log \left[P(z_{t,s} | x_t, m_{c_{t,s}}) \right] \right] \tag{4.5}$$

$$(X^*_{0:T}, M^*) = \underset{X_{0:T},M}{\operatorname{argmin}} \quad -\sum_t \log \left[P(x_t | x_{t-1}, u_t) \right] \; -\sum_{t,s} \log \left[P(z_{t,s} | x_t, m_{c_{t,s}}) \right] \tag{4.6}$$

If we assume that:

- These conditional probabilities are Gaussians,
- The Motion Model is noted f and the motion measurement error covariance matrix Q_t
- The Observation Model is noted h and the observation measurement error covariance matrix $R_{t,s}$
- $|| \cdot ||^2_X$ is the Mahalanobis distance using a covariance matrix X

The above-mentioned equation becomes:

$$(X^*_{0:T}, M^*) = \underset{X_{0:T},M}{\operatorname{argmin}} \underbrace{\sum_t ||f(x_{t-1}, u_t) - x_t||^2_{Q_t}}_{\textbf{Motion Constraints Error}} + \underbrace{\sum_{t,s} ||h(x_t, m_{c_{t,s}}) - z_{t,s}||^2_{R_{t,s}}}_{\textbf{Observation Constraints Error}} \tag{4.7}$$

Consequently, the overall estimation problem can be reduced to a **Non-Linear Weighted Least-Squares Optimisation Problem**.

4.2.3 Different Approaches for a Robust Graph SLAM

One of the limitations of the above formulation is that it only computes an optimum, regardless of the quality of the input data. First, let's mention an important general remark. By using the Mahalanobis distance, each error term is weighted by the covariance matrix of the measurement. This means that the more uncertain the measure is, the lower the weight of the error term for a given deviation to the prediction. In particular, based on a mass-spring analogy commonly used in the SLAM graph community, all happens as if a set of masses (poses) connected by springs (edges) with different stiffness (level of uncertainty coded by the covariance) had to be adjusted. The greater the uncertainty, the more flexible the "spring" is. The desired configuration of weights is the least energy-intensive one.

Similar to a machine learning algorithm where the concept of " garbage in, garbage out " applies, if the initial design of the graph is erroneous, then the result of the optimization will necessarily be inconsistent. In general, this risk is often due to the formation of erroneous LCCs at the front-end part of the SLAM framework. These LCCs can force the search for a map based on an initially false topology. This is even worse in the operational situation considered in this study (and that is one of the reasons why it is an original research topic). In fact, because of the breakdown, a second source of uncertainty impacts the topology of the graph. However, this time, this is at the level of the so-called "odometry backbone" (or motion measurement backbone). In fact, when a good initial approximation of the trajectory is available thanks to the INS, many erroneous LCCs can be discarded easily. Unfortunately, we do not have this "spine" here. For this reason, it is particularly important to look at the methods that make the SLAM graph approaches robust and especially to study the influence of the trajectory interruption on them.

For the sake of simplicity, the robust methods can be divided into two quite general categories: those that act on the cost function by minimizing the part played by outliers, and those that tackle the very topology of the graph and seek to modify it during the optimization process.

With regard to the first category, it consists in influencing the calculation of the error term generated by a measure that diverges significantly from the prediction. In the non-robust case, for any linear increase in this difference, the associated error term varies quadratically. Therefore, in the case of an erroneous LCC, the closer to the "true" topology of the graph, the greater the error. The optimization algorithm is " constrained " to stay in a neighborhood where this term does not become massive. The result is a corrupted map. To deal with this problem, it is sufficient to apply so-called robust cost functions such as Huber Loss function [41]. The objective is then to modify the calculation method when the deviation exceeds a certain threshold. In the case of Huber Loss function, the variation of the error becomes linear after a level set by the user. There is a wide variety of such functions. However, it should be noted that it does not completely remove an outlier constraint. At best, the error term is contained but it does not fully disappear and the erroneous constraint will always influence the structure of the map, at least a little.

One approach to cope with erroneous data association is to develop ways to dynamically change the structure of the graph. These methods became increasingly popular since the early 2010s. So far, the structure of the graph was only decided at the front-end level of the framework. For example, different image matching algorithms could generate associations that seem plausible based on visual descriptors in a vicinity that could be derived from on an initial estimation of the trajectory using INS measurements. The back-end part which deals with the optimization simply returns a simple *maximum a posteriori* on the basis of the data/hypotheses coming from the front-end.

Over the last decade, several authors have been looking for a way to modify the topology of the graph during the optimization phase. The objective is to be able to definitively "kill" the influence of certain associations that seem unlikely because of the consequent discrepancy between prediction and measurement.

A first approach is based on the following idea: how to adjust the influence of a LCC during the optimization process to measure whether its potential "deletion" would produce a better score for the optimized function? In other words, is it possible to deactivate a constraint, within the optimizer itself, that causes an error term so large that it propagates to the surrounding constraints? This is the idea proposed by Sunderhauf through the so called switchable constraints : [95]. To do this, a weight is added to each switchable LCC that varies during optimization between 0 and 1. This weight introduces a new latent variable that is encoded as a switch vertex. In order to avoid the automatic deactivation of all constraints by switching their weight to 0, a regularization term is added to the optimized function. This additional term represents the difference between the current weight of the constraint compared to its original value when the graph is initialized. Thus, varying the weight of such constraint will always have a cost. One of the interesting points of this approach is that when the weight tends towards 1, everything happens as if the constraint is activated, which make it behave like any other constraint in the graph. On the contrary, when the weight converges towards 0, everything happens as if the LCC does not exist. The topology of the graph can then vary and reconfigure itself during the optimization. Finally, one should note that this method allows to go back by reactivating the constraint if deemed relevant.

$$\begin{aligned}(X^*_{0:T}, S^*) \quad = \quad \underset{X_{0:T},S}{\operatorname{argmin}} \; & \Bigg[\underbrace{\sum_t ||f(x_{t-1}, u_t) - x_t||^2_{Q_t}}_{\textbf{Motion Constraints Error Term}} \\ + \; & \underbrace{\sum_{t,s} ||\Psi(\omega_{t,s}) h(x_t, x_s) - z_{t,s}||^2_{R_{t,s}}}_{\textbf{Observation Constraints Error Term}} \\ + \; & \underbrace{\sum_{t,s} ||\omega_{t,s} - \omega_{0t,s}||^2_{\Omega_{t,s}}}_{\textbf{Prior Switch Weight Reg. Term}} \Bigg] \end{aligned} \tag{4.8}$$

Where:

- $\omega_{t,s}$ the current weight of the observation constraints linking poses x_t and x_s. *Note: the above-mentioned formulation applies to a Pose Graph model as landmarks are implicit.*
- Ψ a function applied to the weights that returns a value between [0..1]. Identity Function is preferred over Sigmoids as its derivative is always equals to 1 which eases the optimization process.
- $\Omega_{t,s}$ is a free parameter of the model. It plays the same part as a measurement covariance matrix. It is used to weight the error generated by any difference between the current weight $\omega_{t,s}$ and its initial value $\omega_{0_{t,s}}$. The parameters $\Omega_{t,s}$ and $\omega_{0_{t,s}}$ are set at the front-end on a case by case basis.

A second approach, the MaxMixtures constraints (or MaxMix) [71], is inspired by the mixtures of Gaussian used by some Multi-Hypothesis Tracking algorithms (like in the case of a filtering approach). The aim is to transpose the idea of a multimodal probability distribution to describe the following probability:

$$p(z_{t,s}|x_t, m_{c_{t,s}}) \tag{4.9}$$

In the basic model of the graph SLAM, it is assumed that this probability density function follows a normal law who happens to be unimodal (only one extremum, here a maximum). In the case where there is a high uncertainty on the association, it seems preferable to consider several options and therefore to have a multi-modal distribution. Unfortunately, it is not possible to choose a Gaussian sum for an optimization approach (the formula to be optimized then becomes difficult to simplify because of the log operator). To overcome this issue, the authors define a model based on a max operator:

$$p(z_{t,s}|x_t, m_{c_{t,s}}) = \max_k \upsilon_k \ \mathscr{N}(\mu_k, \Sigma_k) \tag{4.10}$$

Where:

- $k \in [0..K]$ represents a given hypothesis.
- $\mathscr{N}(\mu_k, \Sigma_k)$ is a Gaussian distribution of mean μ_k and covariance matrix Σ_k
- μ_k is a weight allocated to the hypothesis k. For instance, the users applied the error rate of the front-end data association algorithm to weight the association & null hypotheses accordingly.

This modeling maintains a simple optimization calculation since the max operator eventual selects a Gaussian function. An interesting application of this approach is that it is possible to encode several alternatives. In particular, it is possible to both encode the association hypothesis againt the null hypothesis, where the LCC is supposed to be non-existing. The selection of the maximum then acts as a selector of the most likely hypothesis during the evaluation process.

Another interesting alternative is presented through the Realizing, Reversing and Recovering (RRR) algorithm [59] . What distinguishes this approach is that the different hypotheses of data associations are grouped together in the form of clusters. These clusters gather LCCs that act in the same neighborhood of the "odometry" backbone (Note: this term just refers to the set of poses that are estimated only using motion constraints. It is generally used to initialize the poses values during the first iteration of the optimization process). In order to find the correct data associations and therefore the true graph toplogy, the algorithm measures two different types of level of consistency: at the intra-cluster and the inter-cluster levels. During the first phase, the algorithm detects clusters of hypotheses that are not consistent with each other. In the second phase, it measures if clusters that are mutually consistent. Both consistency evaluation are done by running a χ^2 test using the residuals of the optimized function for a graph that only retains the relevant LCCs cluster(s). This method is driven by one of the basic principle of SLAM, which states that true data associations are necessarily consistent with each other, since they reflect the ground truth. Hence RRR's objective to identify a best set of clusters and use them as a baseline to test the others. This search for a sort of consensus is somewhat similar to the RANSAC method, which can also happen to have been applied at the front-end level of the SLAM problem to find the best combinations of hypotheses.

Some ideas related to the RRR approach are interesting with respect to our case study. Indeed, the operational failure creates a form of asymmetry in the graph where some regions are more reliable than others. Therefore, trying to identify clusters of associations in the vicinity of a "reliable" part of the odometry backbone is an interesting perspective. These three methods have been benchmarked within [96]. It is difficult to claim that there is an obvious "winner" among these algorithms. Indeed, they have different parameterizations and offer relative results that vary from one dataset to another. On the other hand, it is clear that all these methods remain vulnerable to the quality of the initial trajectory estimation based on the model motion and inertial measurements.

Worth mentioning is also the work of Pfingsthorn and Birk [77], who use the concept of hyperedge to connect one single pose to many other vertices. The authors introduce a middle step between the front-end that creates these hyperedges, and the back-end that performs a classical unimodal optimization of a factor graph. This intermediary stage is called "Generalized Prefilter". It searches for the correct topology by generating a spanning tree that unrolls from leaf to leaf a new unimodal graph topology for each association option encoded in each hyperedge. Each association option can be multimodal by nature, similar to what is developed within the MaxMixtures.

4.3 Application to an AUV mapping an area using a SSS (Ground Truth)

In this section, we propose to use the above-mentioned graph SLAM model to generate a mosaic from successive sonar images from our dataset. This survey is made of several independent tracks with overlapping regions that were acquired by a side scan sonar mounted over a towfish. As described in Chapter 2, this is an underwater payload that is towed by a surface ship using a flexible cable. In terms of available measurement, we have inertial data that are acquired by the INS of the towfish and GPS coordinates of the surface ship. Because of the flexible link, it is not possible to derive accurately the sensor absolute location. Furthermore, the fish velocity and altitude are measured via a DVL.

Last but not least, there is no data acquired during the inter-track phases. The only information about the relative positioning of these tracks comes from the surface ship GPS data (Figure 4.5). In our experiments, we derive an AUV survey scenario by ignoring the information related to the surface ship. The INS measurement of the fish are sufficient to model it as an AUV operating alone. The lack of information during turn is interpreted as the operational failure that disrupts the continuity of the trajectory estimate. The map is therefore generated using information extracted from the images.

In order to build the graph of constraint, we process the INS data and produce a 2D trajectory (basically the nadir of the sonar system) as described in Figure 4.3. We use an EKF filter coupled with a constant velocity model as described in [18]. The objective is to simply produce a first estimate of the trajectory, that will be corrected by the SLAM algorithm later. One of the challenges of the dataset is related to the altitude measurement that occasionally contains erroneous values as presented in Figure 4.4. The filtering phase helps in mitigating that issue and produces a smooth variation of the altitude. Note: If there was a seafloor anomaly responsible for a sudden altitude shift (like a rift...) this should be observable on the sonar images, however, this is not the case with the dataset.

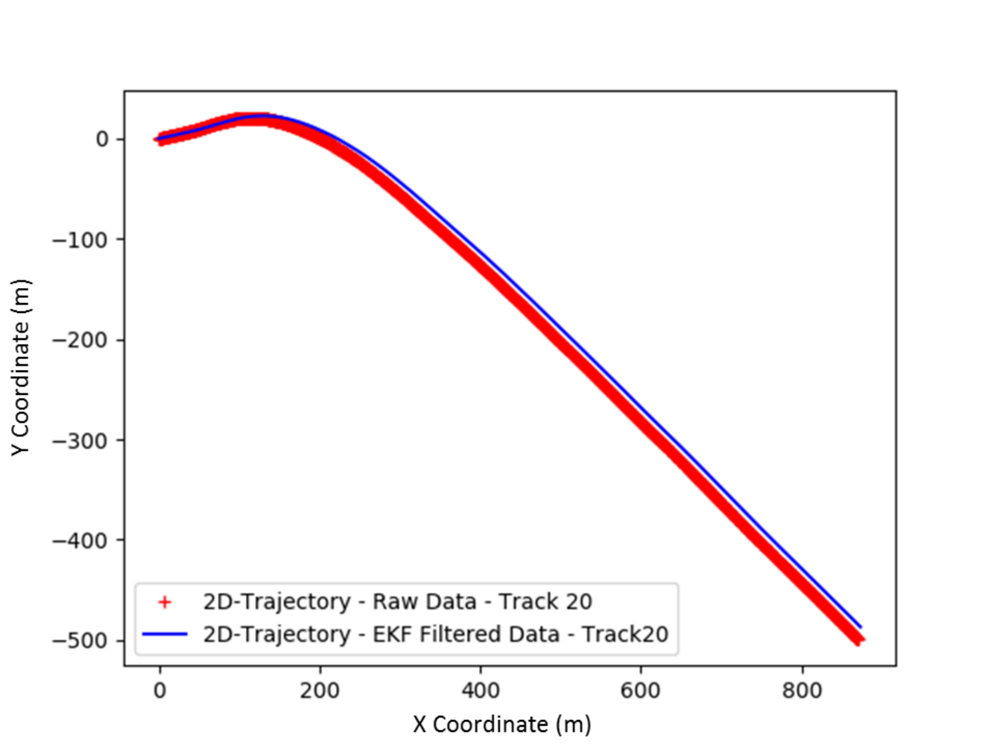


Figure 4.3: 2D Trajectory Estimated via an EKF and a Constant Velocity Motion Model similarly to [18]

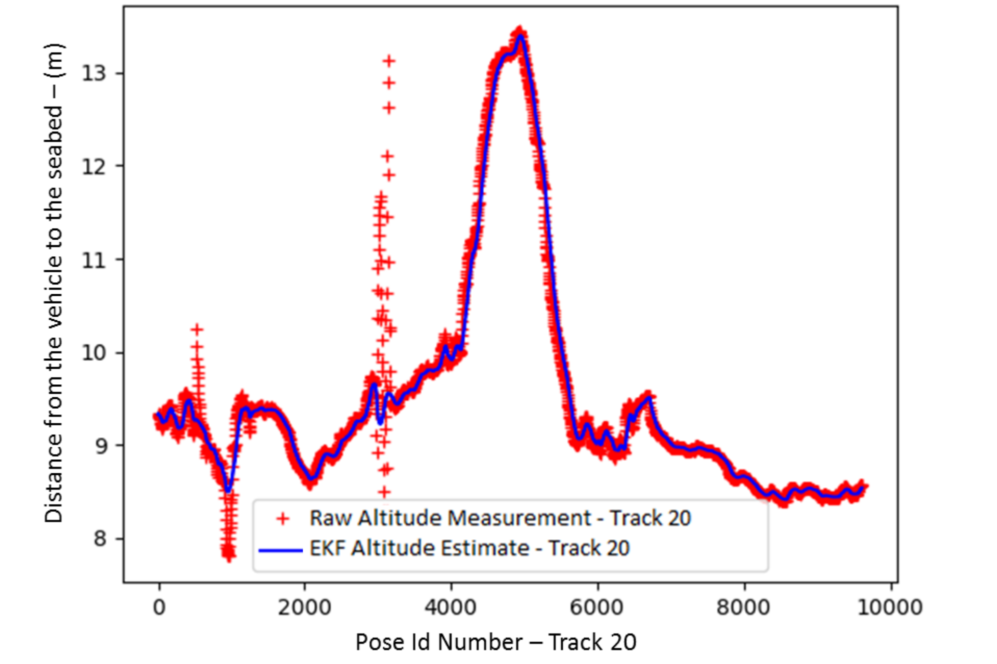


Figure 4.4: Altitude estimate using DVL measurements.

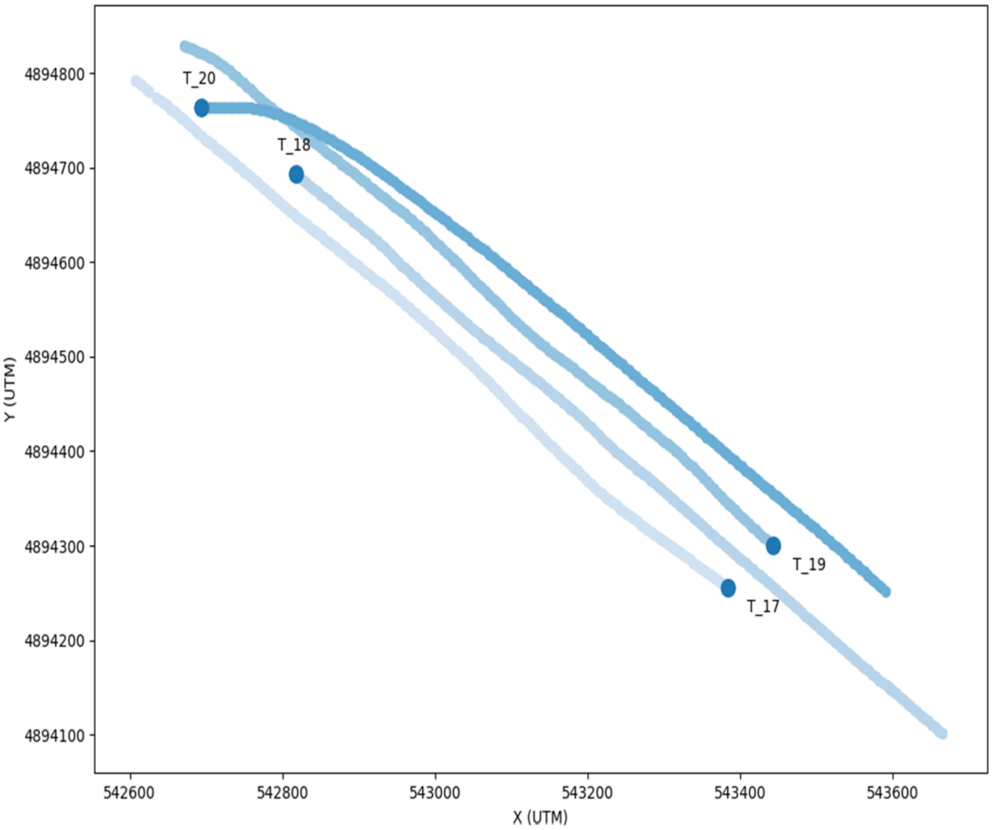


Figure 4.5: Surface Ship GPS During the Survey

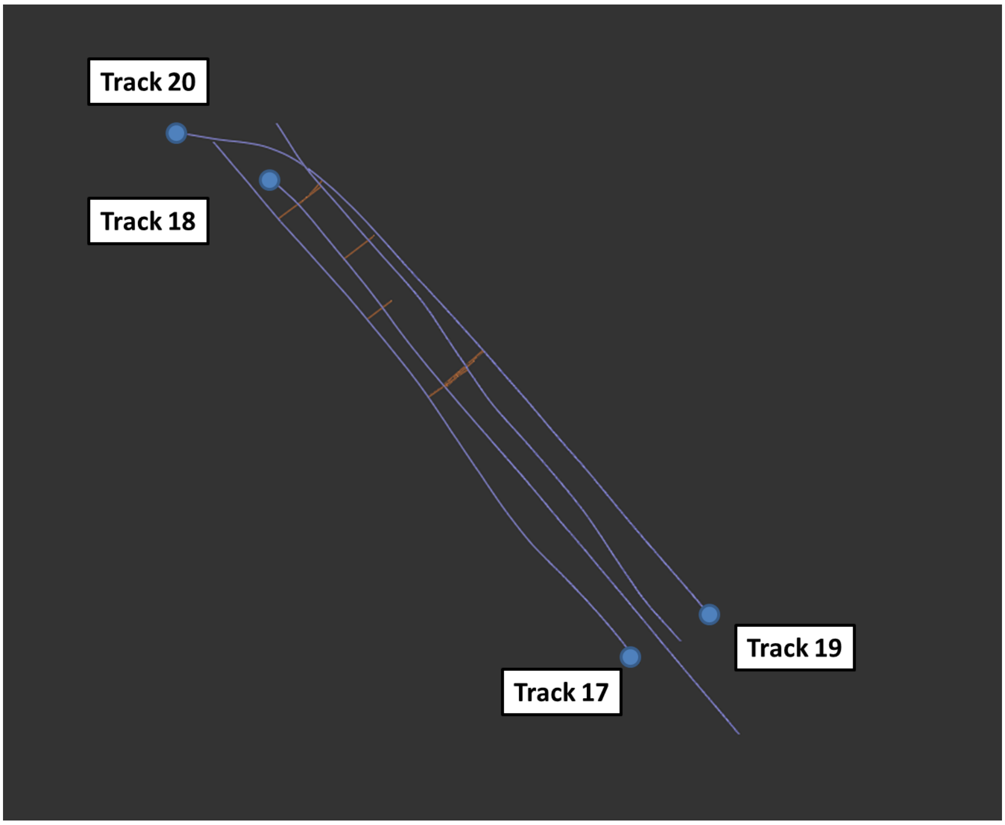


Figure 4.6: Post-Optimization Graph using Perfectly Paired Landmarks

In our experiments we used the G2O optimizer [36] that happen to be a popular choice for generating graph SLAM scenarios [35]. There are other alternatives like iSAM 2 [47], however G2O has the advantage of being compatible with other SLAM tools used in this study. Furthermore, it presents the advantage of being customizable: one can define new types of vertices and edges to model more complex relations.

The first step in building the graph consists in defining the motion constraints. In this 2D-model, they are encoded as a SE2 transformation between successive poses (basically a translation and a rotation in the local frame) . The above mentioned trajectory is used to produce these constraints. The second step requires to add the observation constraints. These constraints are derived from data associations extracted from the sonar images. There are various ways to encode observation constraints. In Figure 4.6, we show a landmark-based graph where orange constraints represents the spatial relation between the vehicle pose and a landmark defined by its Cartesian coordinates. Such constraint is later referenced as a SE2-XY edge: SE2 because of the vehicle pose, and XY because of the landmark pose. We simply inserted an XY-XY edge between each pair of landmark, set to a distance of $0m$ to encode the matching hypothesis. The SE2-XY constraints parameter are simply derived from the slant range corrected acquisitions under the flat bottom hypothesis. The blue part of Figure 4.6 represents the vehicle poses and the motion constraints. When comparing the optimized graph with the surface ship GPS coordinates, it is interesting to note an offset at the extremities of each track. This gap is in fact the distance difference due to the flexible link between the surface ship and the sensor.

Once the post-optimization AUV (or towfish...) trajectories and derived by G2O, we project the each bin of each ping using a rectangular approximation (with across-track varying shape to model the change of resolution). Several georeferenced sonar tiles are generated and merged to form track images. These tracks are then combined to create a survey mosaic. Figure 4.7 represents the merging of 3 out of a 4-tracks survey (for the sake of clarity) using the maximum intensity value of each image when creating the mosaic. On the other hand, Figure 4.8 uses 3 different color channels to allow a better visualization of the overlaps. They are different ways to process overlapping pixels and they are generally application-dependant.

The two following pictures 4.9, 4.10 present a detail of the aforementioned mosaic. The multi-channel one offers an interesting example. The Observation Constraint (or LCC) that is present in the region is the one that we identified previously in the Chapter about sonar image analysis (as marked by the yellow cross). As we can see on the lower part of the merged image, the quality of the registration decreases when we move far from the origin of the Observation Constraint. It means that another step of finer registration, using traditional methods, should be applied to improve the quality of the merged product. **It also highlights the fact that observation constraints exercise a local influence**. Overfitting observation constraints may negatively impact the low-level quality of the mosaic.

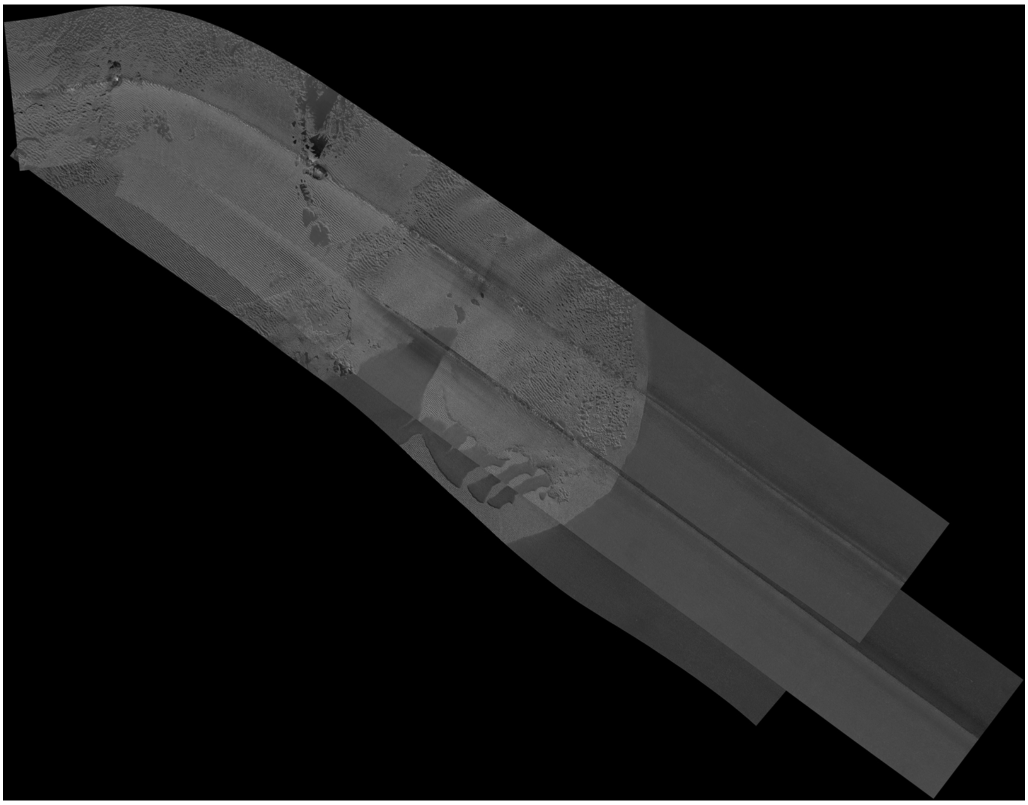


Figure 4.7: Side Scan Sonar Survey Mosaic: maximum values selection

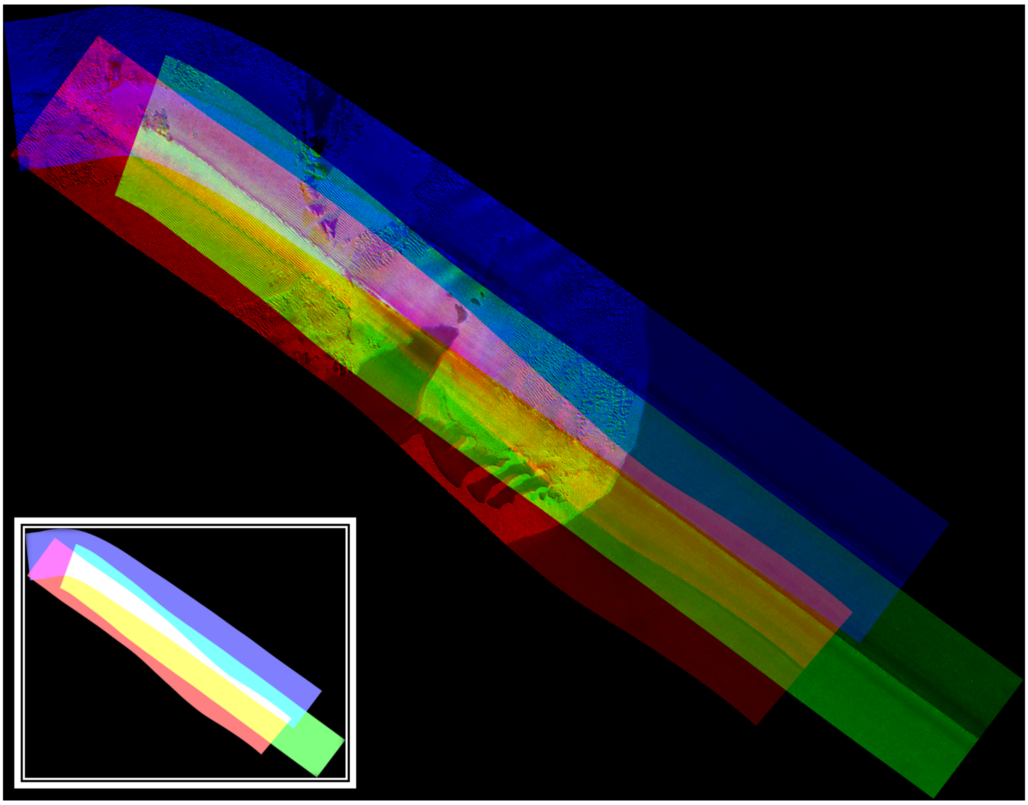


Figure 4.8: Side Scan Sonar Survey Mosaic: multi-channels overlap

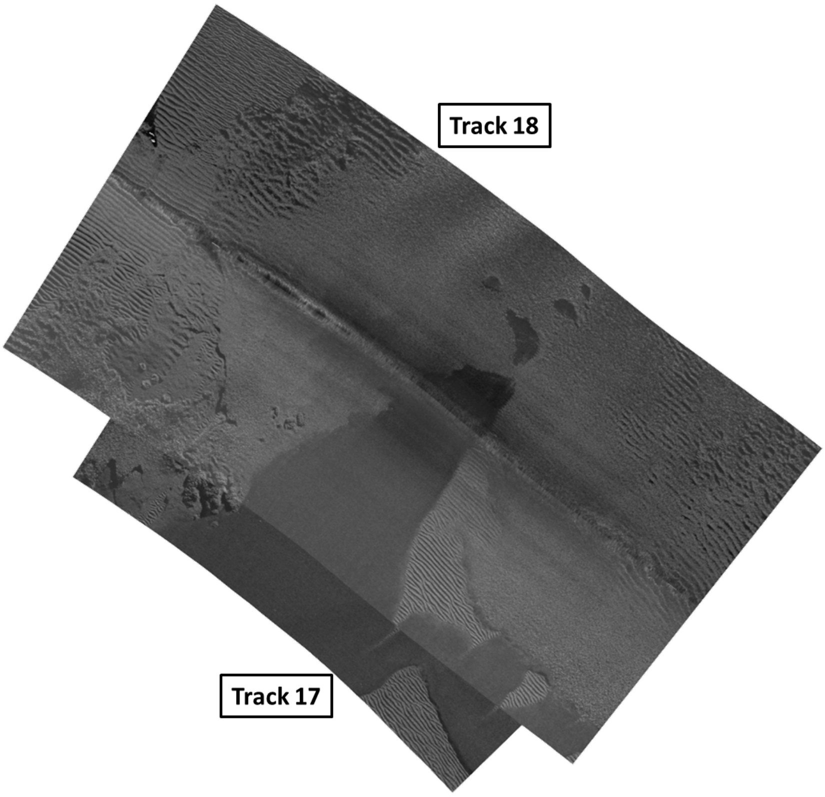


Figure 4.9: Mosaic Detail: maximum values selection

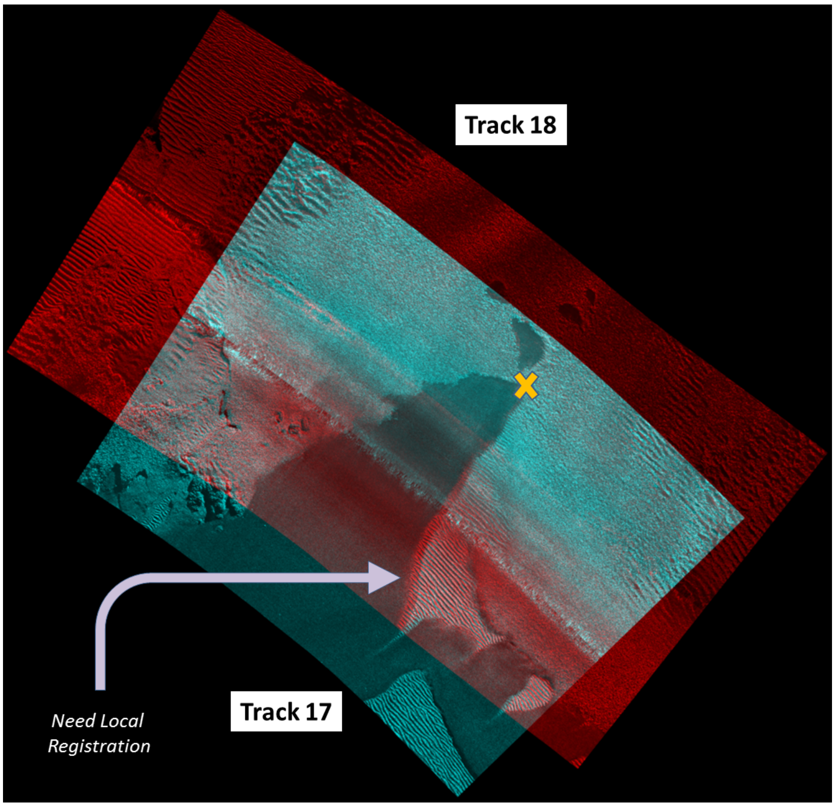


Figure 4.10: Mosaic Detail: multi-channels overlap

4.4 Case Study: Graph SLAM in presence of LCCs outliers & temporary vehicle breakdown

After producing a mosaic using perfectly associated landmarks, the effects of Observation Constraints (or LCC) outliers have to be assessed. In order to do so, we formed a set of 18 matches containing:

- 3 Perfectly Associated Points
- 3 Outliers in the vicinity of the correctly associated observations (and based on visual similarity): they model data association that could result from pairing look-alike points.
- 6 Additional Outliers resulting from random associations of very distant points: they model highly abnormal associations.

In this scenario, we just use two successive tracks. We suppose that we have no to little information about what happened within the inter-track region to model the breakdown scenario. The objective is to get an appraisal about the behavior of the optimized graph depending of the number of outliers, their nature and the hypotheses about the junction between the two tracks.

Figure 4.11 highlights how the optimization produces an inconsistent map in presence of a large number of outliers when no robust modelling is applied to reject outliers. The left images represent the scenario where there is no constraint to link both groups of motion constraints from the different tracks. On the right, an additional constraint has been introduced at the inter-track level to encode the rotation between both track using for instance a measurement that could result from a compass. While this constraint is very "flexible" it constrains the topology of the scene a little bit. This constraint will be particularly useful for the next tests. Furthermore, every observation constraint has been encoded using XY landmarks like in the previous section.

In the next batch of experiments, we apply switchable constraints to the LCCs in their "vanilla" implementation as described in [95]. As previously explained, the optimizer can turn the constraint "off" by adjusting its weight to 0 if it is more beneficial for the cost function. Reminder: there is also a cost for deactivating a constraint in order to avoid deactivating all switchable LCCs. After optimizing the graph, only a group of mutually coherent constraints emerges as it is less expensive to deactivate all other ones, rather than impacting too much the motion constraints (or the "backbone"). As we have seen in the part about the theory of SLAM, a property of correct data associations is that they are mutually consistent since they represent the ground truth. However, it does not mean that during optimization false constraint randomly prevail and force the deactivation of some correct constraints: especially if these prevailing outliers are in the vicinity of the correct associations in the graph topology.

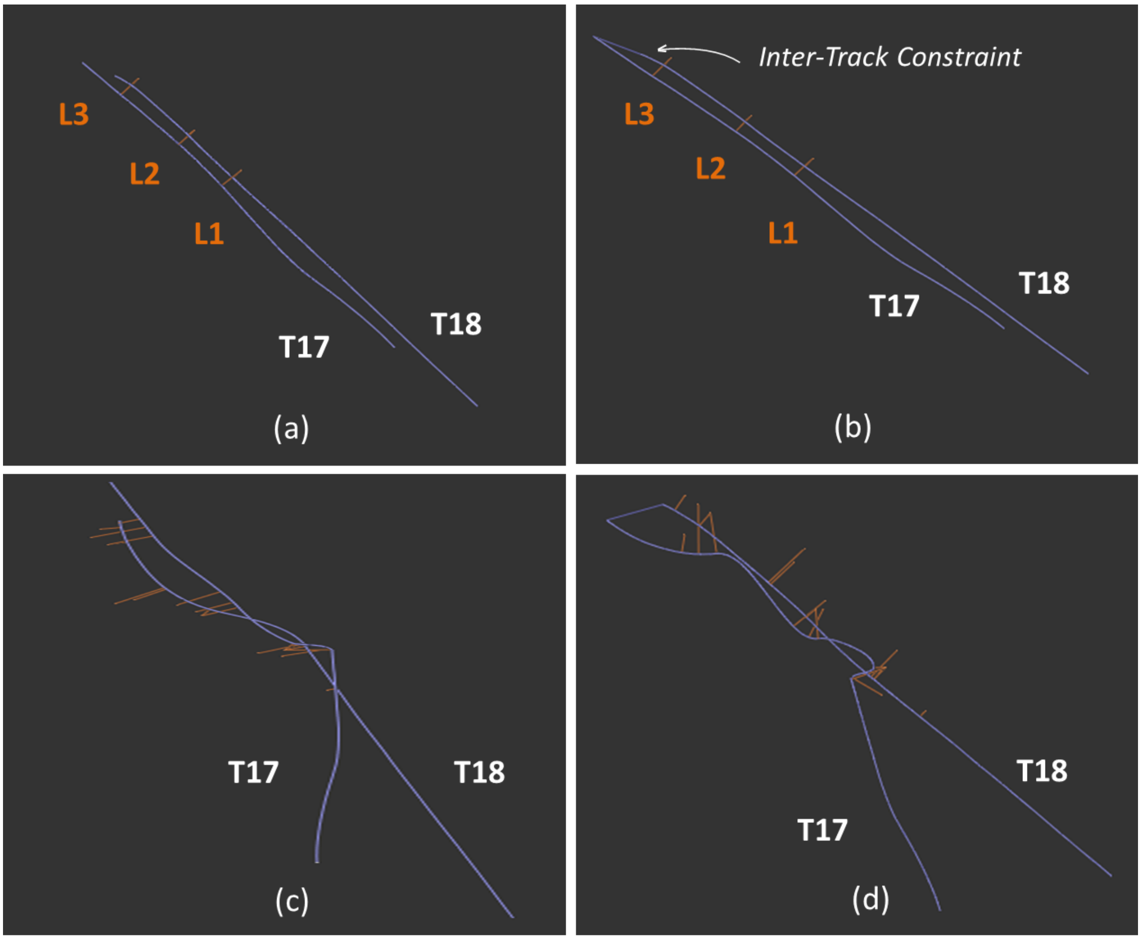


Figure 4.11: Evolution of the Optimized Graph when Erroneous LCCs are introduced without robust modelling.
(a): Perfect Matches, No Inter-Track Constraint
(b): Perfect Matches, Inter-Track Constraint
(c): Including Outliers, No Inter-Track Constraint
(d): Including Outliers, Inter-Track Constraint

This is why we separated our outliers into two categories. The first group contains outliers that model local association errors: they do not tend to alter too much the ground truth graph topology. Therefore, they would result in trajectory corrections that are not as significant as what grossly abnormal associations would suggest. These later type of outliers constitutes our second group. The true location difference between the two points that were incorrectly associated is so high that the post-optimization topology of the graph would be dramatically bent by these constraints, if prevailing. One of our objective is to assess how switchable LCCs behave in presence of a mix of correct matches, local erroneous ones and grossly abnormal ones. In Figure 4.12, the first scenario (b) presents all the matches tested at once while (a) is the ground truth using correct matches only. We note that, while the abnormal constraints are deactivated, some local erroneous LCCs prevail and alter the topology. It can be verified in part (*c*) where only true LCCs and the erroneous ones located in the vicinity are tested. The resulting topology is very similar from the one in (*b*). On the other hand, in the case of (*d*), we can see that mostly grossly abnormal LCCs were introduced, the switch constraints easily get ride of most of them.

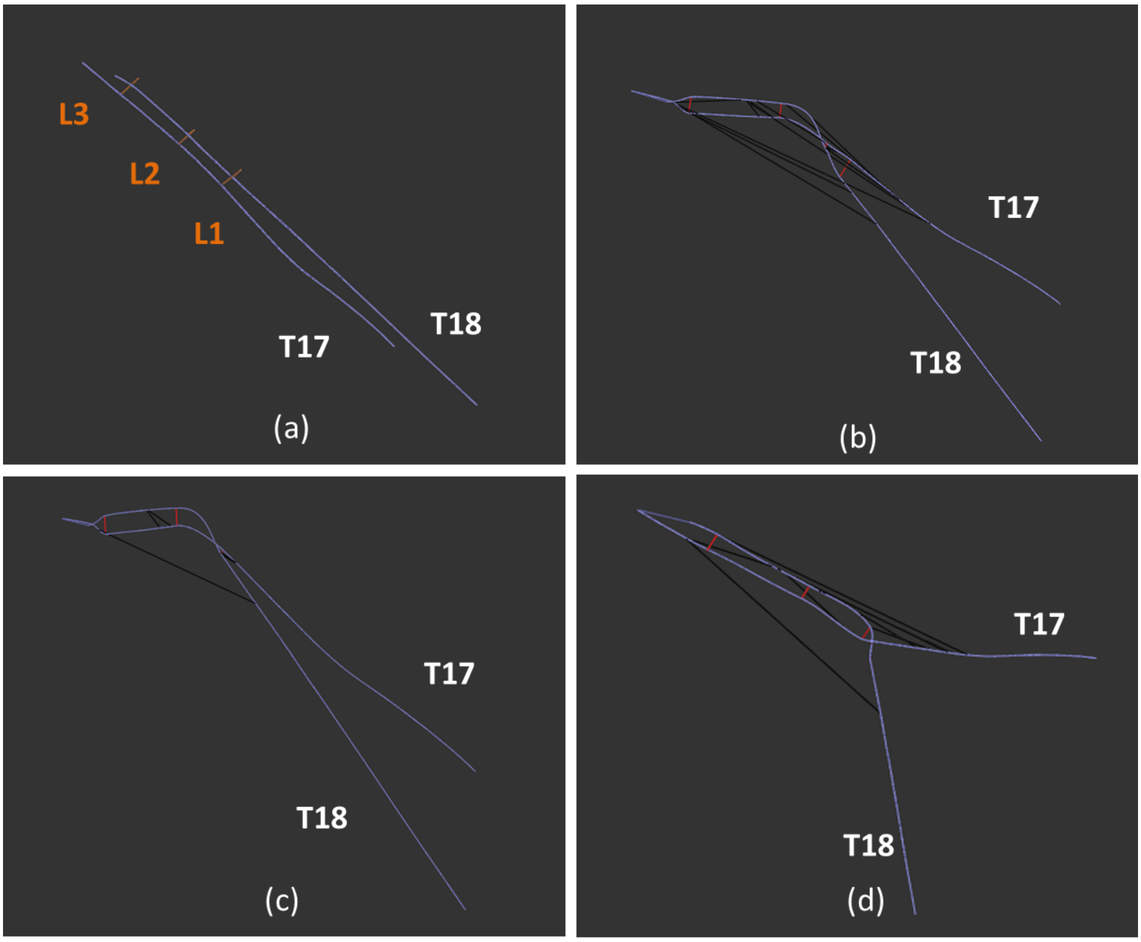


Figure 4.12: Using Switchable Constraints to Encode LCCs
(a): Perfect Matches Only (Baseline)
(b): All Matches
(c): Correct Matches + Erroneous Ones in their Vicinity
(d): Correct Matches + Grossly Abnormal Ones

In fact robust optimization-based SLAM is very dependant on the accuracy of the motion constraints (ref: the infamous "Odometry Backbone"). Unfortunately, the lack of information with the inter-track region is a potential source of error that may compromise the topology resulting from such type of approaches. In the last example, we can see that knowing at least one correct LCC may be enough information to deactivate most or every outliers 4.13. This simple example highlights that if we manage to confirm, *a posteriori*, that some specific data association were correct, we can then revisit the optimization phase and force these constraints to play a prevalent part (either by increasing their level of certainty, or conversely reducing their measurement covariance).

Therefore, the resulting post-optimization topology would only retain LCCs that are consistant with these "certain" associations. We could also use these few *a posteriori* validated LCCs to infer some information about the inter-track region and improve its parameterization in a new graph. This observation suggest a critical idea: it may be possible to iteratively converge towards the ground truth by leveraging this interesting property of robust opmization-based SLAM. Last but not least, we had to add an inter-track constraint in order to keep the optimizer stable: as all the LCCs can be deactivated, the graph topology could become fully disconnected, which crashes the optimizer.

In a previous study, we added several fictional poses in the inter-track region, connected by flexible motion constraints (with a high covariance) [43]. If we could find a parameterization of these modelling poses that reflects the physical limitation of the vehicle motion during the breakdown, we could constrain the search to a more reliable topology.

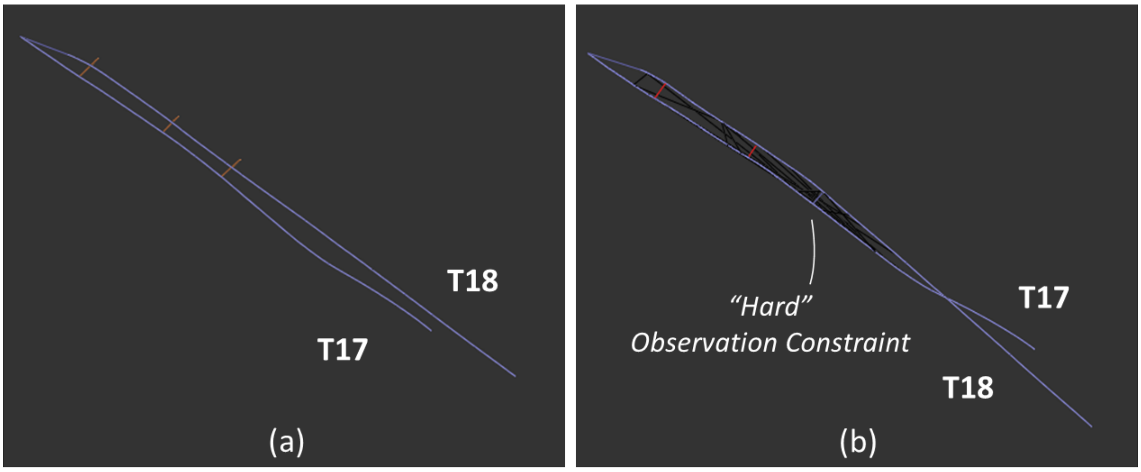


Figure 4.13: Effect of setting one correct LCC over all erroneous ones.
(a): Ground Truth
(b): One Correct LCC is encoded as "certain" in the set of observation constraints. Consequently, most of the erroneous ones are deactivated.

4.5 Conclusion

The different experiments we made, coupled with the definition of our operational scenario, suggest that an *a posteriori* validation of data associations will be necessary to confirm or reject the previously associated landmarks. Furthermore, we also note the importance of finding a good approximation of the true topology to constrain the search of the optimized trajectory in a plausible region. The framework that addresses this problem is presented in the following Chapter.

To further close the loop on the validation phase, the obvious tool to do it will be the sonar mosaic itself. However, while so far we were dealing with a global registration problem to build a first graph topology, after optimization, the tracks are approximately registered (correctly... or not!). That means that we can rely of image processing methods that leverage the fact that, assuming the LCCs are correct, the overlapping acquisitions should match over a large scale. We can therefore envisage to deploy tools that perform macroscopic image validation rather than trying to associate discrete pair of points (as for the LCCs).

If such mosaic analysis tend to produce inconsistent results in a specific vicinity, it may be related to the presence of a nearby false LCCs. Conversely, if two regions overlap particularly well, in that case, it could mean that the local LCCs are correct, therefore we could iterate the graph topology search by forcing these LCCs to prevail.

Chapter

5

A Robust SLAM Framework for Contested Environments

Highlights of the Chapter

- Contemporary navies are now moving towards long-range and discreet autonomous systems that operate in harsh environments. The stealth requirement raises a particularly serious challenge for autonomous navigation as the system cannot surface to capture a GNSS signal. Even if surfacing were an option, the possibility that an adverse party is spoofing the signal should be taken into account. It is even more complex if the AUV is deployed in an area that has not been previously mapped. Through the presentation of the literature review and different experiments, the previous chapters have highlighted the vulnerability of such autonomous systems when it faces a critical situation and gets lost. Its only option remains to relocate itself relying on its past and present acquisitions, which is not simple due to the characteristics of the acoustic lateral imager and the nature of the failure.

- The crucial point of this thesis work was therefore to put in place a strategy to address this problem. The most important constraint that influenced the formulation of this approach was to be able to deal with a global relocation problem in a context where data association errors are likely to be numerous. It is therefore a question of being able, in a first step, to determine the correct topology of the scene. To achieve this, we have used and further adapted the graph-oriented methodology presented in the previous chapter by developing a set of tools to deal with the above-mentioned operational scenario.

- First, this chapter details the constraints and motivations taken into account in the design of our framework. Second, we detail step by step, the structure of the data fusion algorithm that has been developed during this thesis, starting from the raw data to a georeferenced sonar mosaic with several layers of information. Moreover, the tools that have been developed at the level of the SLAM algorithm are explained in order to test several types of associations to be able to adapt to different observation modalities of the environment.

- In order to be able to determine the correct topology of the scene, the data fusion algorithm has been developed in such a way that it is possible to finely study different parts of the graph and the mosaic. It is then possible to measure the effect of an observation constraint or a degradation/loss of the vehicle motion signal. The framework allows to test different tracks connection hypotheses that can be used as a very coarse modelling of the vehicle pathway during the breakdown. The overall objective of this algorithm was to operate a coupling between the *a priori* image study, that informs the initial structure of the graph, its post-optimization analysis, and the plausibility of the a posteriori mosaic. The issue of validation or rejection of an association is an important point of this part. Indeed, in the case of a global relocation problem, it is possible that the connecting trajectories are grossly erroneous if some false data associations have prevailed during the optimization of the graph.

- The results are presented according to different operational situations, more or less deteriorated, and the evolution of the parameters that characterize the graph and the mosaic. The strengths and limitations of this methodology are discussed based on different examples, however, we open the way to future perspectives that would help overcome some of the problems encountered. In particular, the use of heterogeneous observation sensors would address some of the limitations of the sole use of a side scan sonar. Fortunately, since the core of the Framework is agnostic to the modality of the sensor used, it would indeed be possible to apply this approach to a situation where a bathymeter is also available, for example. Another point of improvement would also be the use of more advanced techniques to estimate the trajectory at the time of the failure, by further restricting the likely poses by delving into the mechanical properties of the vehicle and hydrodynamics.

5.1 Strategy And Motivations

The capability target related to this study is to design an underwater autonomous navigation system that is resilient against possible operational damages. Furthermore, it is assumed that the environment is potentially hostile, which excludes any GNSS repositioning. Finally, it is considered that there is no prior representation (i.e. a map) of the area.

This scenario could be applicable to autonomous scouting missions where discretion is required and it is paramount to not risk losing the system within the operation zone as a result of a breakdown. Indeed, an abandoned AUV in a contested area could be seized by a hostile force, which would lead to increased tensions in the region.

As detailed in the previous chapters, our aim is to allow the repositioning of the vehicle thanks to a map elaborated from the data produced during the mission using remote sensing instruments, in this case an active sonar sensor. Therefore, this poses a problem of global registration of the acquisitions, with a significant uncertainty on the relative positioning of the sonar tracks.

This problem can be addressed using a SLAM algorithm, provided that it can be properly initialized with the right data associations. It is possible to use different methods that may reject some of the hypotheses resulting from the association phase as seen above. However, since graph SLAM models are based on the calculation of a maximum a posteriori, the initialization of the scene topology (and therefore the definition of the function to be optimized) plays a critical role. The greater the initial uncertainty about the relative positions of the tracks, the greater the risk that erroneous observation associations will prevail during optimization. The optimized graph corresponds only to the most "likely" topology, conditionally to the initial hypotheses. However, this does not guarantee a representation close to the ground truth: it may result in an inconsistent map.

This leads to the following situation: not only the front-end part risks to introduce false positives, but also the back-end part, which is based on a MAP approach that can cause the right association hypotheses to be rejected if the uncertainty on the trajectory caused by the failure becomes too great.

One could then be tempted to move towards a methodology based on particle filtering: indeed, it is inherently multi-modal and allows maintaining a large number of possible trajectories. It is an approach widely used in the context of the "stolen robot" problem. However, given the operational context that is considered here, two major issues arise. On one hand, the failure creates such a large initial uncertainty that a very large number of particles would have to be processed, which would make the estimation of the map very cost-intensive. On the other hand, in order to make a particle filter converge quickly, one must be able to make either numerous (and concordant) associations or very discriminating ones. For instance, successive observations of the same point in the environment (such as a wall in an indoor environment), is a rich source of information that allows to quickly narrow the range of likely trajectories. However, the underwater environment is poorly structured and moreover, because the observations are made in the across-track direction, the system must maintain a nearly straight trajectory to avoid coverage gaps that would makes the description of features inaccurate. Consequently, the same point will be observed only once per pair of tracks. Furthermore, the PF has a significant drawback compared to a graph approach. It is a Bayesian filter, i.e. once it has "absorbed" the information from a data association, it cannot be reversed. And this can turn into a major complication, especially if incorrect associations are produced at the beginning of the linking of the two trajectories. The advantage of the graph is that it can be easily reconfigured in real time. By simply removing or adding edges, the MAP can be recalculated using a new topology. The various examples presented in the previous chapter illustrate this principle.

In fact, an interesting idea from the tests on the different types of constraints is the concept of local predominance. As this is an optimization problem, which can be compared to minimizing the energy of a mass-spring system, the algorithm will try to produce the most stable configuration and disable the constraints generating too much "pressure". When considering a group of constraints identified in the same vicinity, two possible scenarios may occur. Either they all are concordant, in which case they "survive" the optimization phase, or alternatively, a subgroup predominates and leads to the deactivation of the other constraints since they correspond to the ground truth. It is critical to emphasize that correct data associations are always concordant with each other. The question is therefore whether or not the optimization has selected a subset of “correct " constraints. The resulting mosaic must therefore be investigated and its plausibility assessed in the predominating constraints neighborhood.

Another important point is that across the entire acquisition scene, correctly and falsely paired regions can be seen to coexist as a result of the optimization. Indeed, an observation constraint only plays a local role within a trajectory. This is particularly true as the inertial unit is relatively accurate over short time intervals. This observation suggests the following approach: if one is able to detect and correctly match a sub-region, then it is possible to relax all the other matching constraints of the graph. This new region can then serve as a hard anchor for the scene topology. This brings the problem back to a traditional local registration problem, where the relative positioning of the tracks is approximately known.

Consequently, the graph approach proves to be rather interesting: one can iteratively refine the topology of the scene in order to get back to the local registration problem, where many tools work. Therefore, it is necessary to build a tool that can be used to define a neighborhood and evaluate the plausibility of the overlapping of acquisitions in this region. However, this principle of an a posteriori validation phase must be based on data processing tools that are different from the ones used for the data association phase in order to avoid redundancy. This is where heterogeneous data fusion may play a decisive role. Such heterogeneity can be of two different forms: either it refers to the active sensor modality (two different types of sonar are used), or by the type of information that is extracted from raw measurements, a bit like in the case of so-called iconic or symbolic approaches. As a result, we have developed a framework that relies on diverse indicators for assessing the likelihood of the resulting mosaic.

A first point of focus is to measure the influence of an observation constraint on the distortion of the surrounding edges in its vicinity. From an image analysis perspective, two complementary criteria can be used: measuring the consistency of the seafloor texture types that are overlaid in the overlapping areas of the different tracks (a key macroscopic criterion) as well as generating iconic heat maps using different similarity measures (a criterion that is related to the underlying distribution of probabilities of the measured signal). While each of these criteria picked individually are not sufficient for solving the general problem, altogether they are useful for rejecting suspicious data association hypotheses and help in relaxing the graph topology. Consequently, only loop closing constraints that have passed both phases are kept in the final graph structure. The aim of this approach is to end up with a map that is a coarse approximation of the actual ground truth.

Because of the nature of the initial level of uncertainty, an iterative approach is necessary. As such, a first set of registered mosaics has to be produced. It is critical to flag this point due to the waterfall mode of acquisition of lateral active sonars. Altering the trajectories affects the projection of the sonar images. An abrupt change in the vehicle's heading can distort a texture or the shape of a remarkable landmark.

There are four thrusts that drive the definition of our approach:

- The ability to implement robust constraints
- The ability to develop different types of matches that are more suitable for diverse types of features. For instance orientable landmark as SE2 poses for relatively flat feature in order to recover some information about the heading, and XY landmark to model future bathymetric point or echo-shadow complexes using XYZ coordinates for a future 3D implementation of the Framework. the aim is to make the framework heterogeneous-sensors ready.
- The ability to easily apply different models to the inter-track breakdown, based on different assumptions and recover the data after information.
- The ability to focus on subsection of the graph, keeping track of each match independently to be able to analyze the post-optimization vicinity of the match to eventually revisit its plausibility.

The structure of the developed framework is presented in the following section.

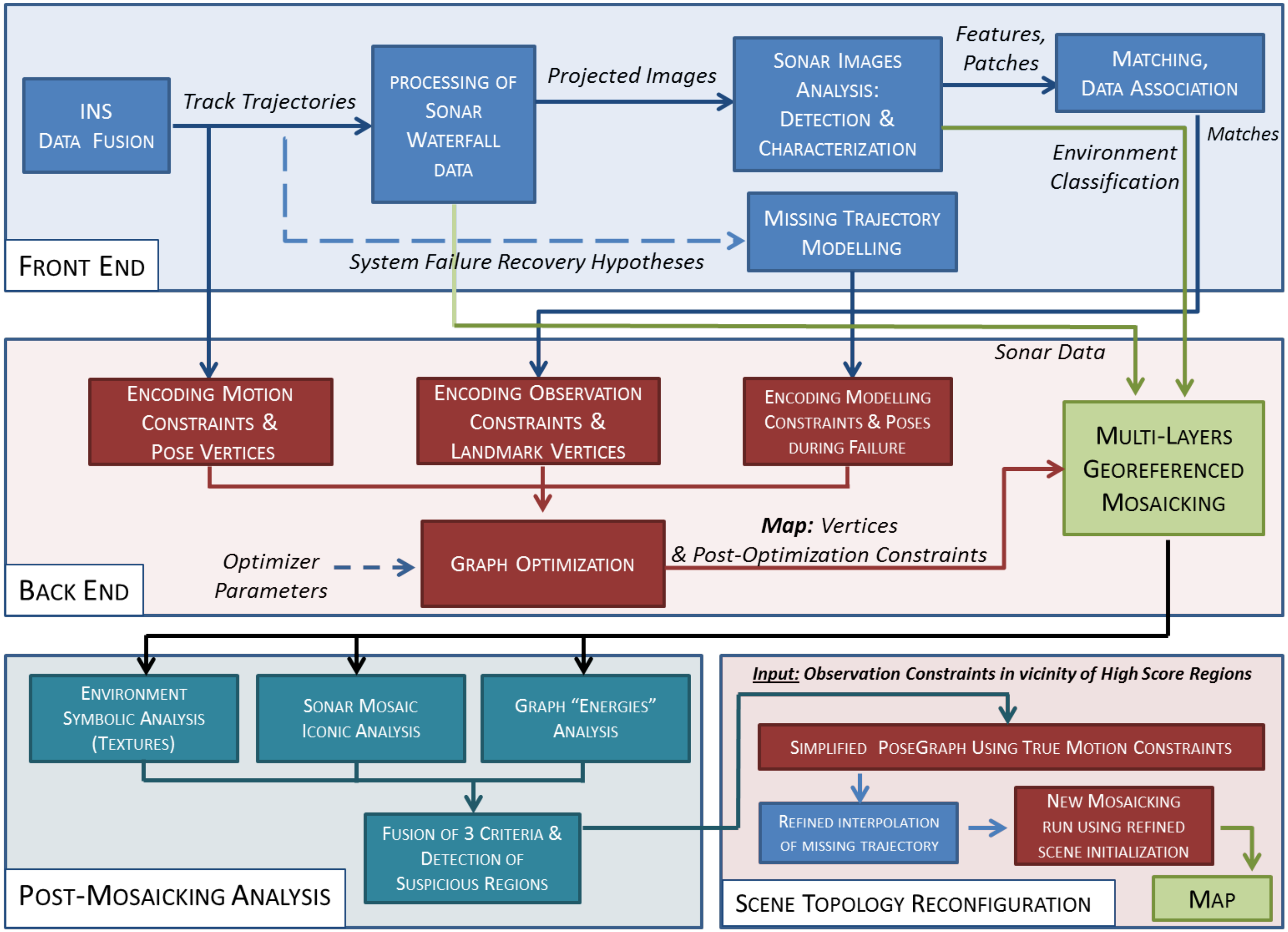
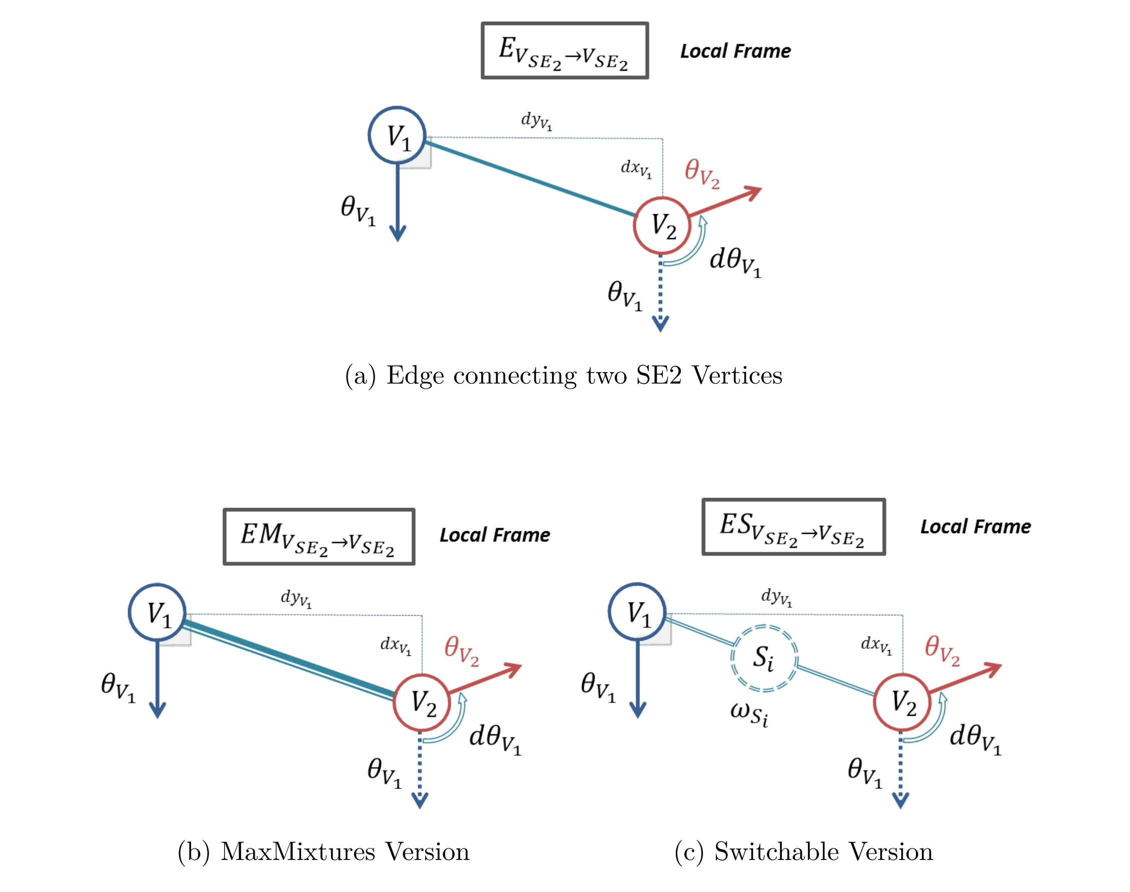

INS
Data Fusion
Track Trajectories
Processing of Sonar Waterfall Data
Projected Images
Sonar Images Analysis: Detection & Characterization
Features, Patches
Matching, Data Association
Matches
Environment Classification
Missing Trajectory Modelling
System Failure Recovery Hypotheses
Front End
Sonar Data
Encoding Motion Constraints & Pose Vertices
Encoding Observation Constraints & Landmark Vertices
Encoding Modelling Constraints & Poses during Failure
Multi-Layers Georeferenced Mosaicking
Map: Vertices & Post-Optimization Constraints
Optimizer Parameters
Graph Optimization
Back End
Environment Symbolic Analysis (Textures)
Sonar Mosaic Iconic Analysis
Graph "Energies" Analysis
Fusion of 3 Criteria & Detection of Suspicious Regions
Post-Mosaicking Analysis
Input: Observation Constraints in vicinity of High Score Regions
Simplified PoseGraph Using True Motion Constraints
Refined interpolation of missing trajectory
New Mosaicking run using refined scene initialization
Map
Scene Topology Reconfiguration

Figure 5.1: Block Diagram of the Proposed Framework

5.2 The Proposed Framework Structure

5.2.1 Overall Presentation

Figure 5.1 presents the rational of our proposal. We aimed to develop a tool that can be applied to different types of sensors and data processing methods. Therefore, we focused on developing an agnostic algorithm that starts from a set of observation constraints that are computed at the front-end (and that are sensor/method dependant). The aim is to easily plug state-of-the-art specialized algorithms to study their influence over the graph topology recovery. The next sections present in detail the path from the formation of various types of matches that reflect the diversity of the underwater environment, to the post-mosaicking tools that have been implemented to check the consistency of the scene representation.



(a) Edge connecting two SE2 Vertices

(b) MaxMixtures Version

(c) Switchable Version

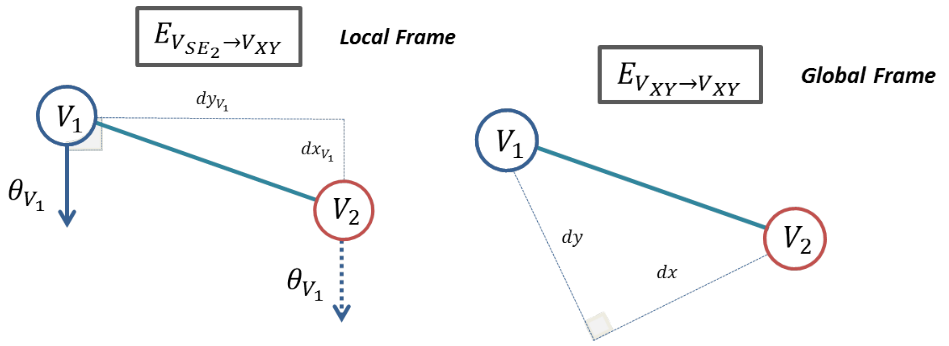
Figure 5.2: SE2-SE2 Edges Used in The Graph . For the MaxMixtures and Switchable versions, we used the implementation presented by their authors.

5.2.2 Types of Edges and Matches

As we are using a 2D model, the graph is composed by SE2 (x,y coordinates & heading) and XY (x, y coordinates) vertices. Therefore there is only a limited number of elementary constraints that encode the spatial relation from vertex to vertex. They are described in Figures 5.2 and 5.3. The parameters of these constraints are either derived from the motion and observation models, in particular the covariance matrices that reflect the level of certainty associated to the constraint as presented in the SLAM chapter.

The values of these variables play a critical part in the optimization process as they command the level of flexibility of each edge, therefore, they have to be derived carefully. Regarding the observation constraints, it is very image & sensor processing driven. For instance, the flat bottom hypothesis used to project sonar images introduces uncertainty that contribute to increase the covariance, thus making the constraint more flexible. Regarding the motion constraints covariance, it is directly related to the uncertainty of the Inertial Navigation System. Since we made several simplifying assumptions so far, we keep it simple as well here by using the mass-spring analogy.

We tested different relative orders of magnitude for the different covariance matrices (for the motion constraints, observation constraints, the inter-track region etc ...). The objective is to gain a qualitative insight about how the whole graph is behaving depending of the variation of uncertainty in these different regions of the graph, especially the effects of the operational breakdown that is therefore modeled as a very "flexible" region.



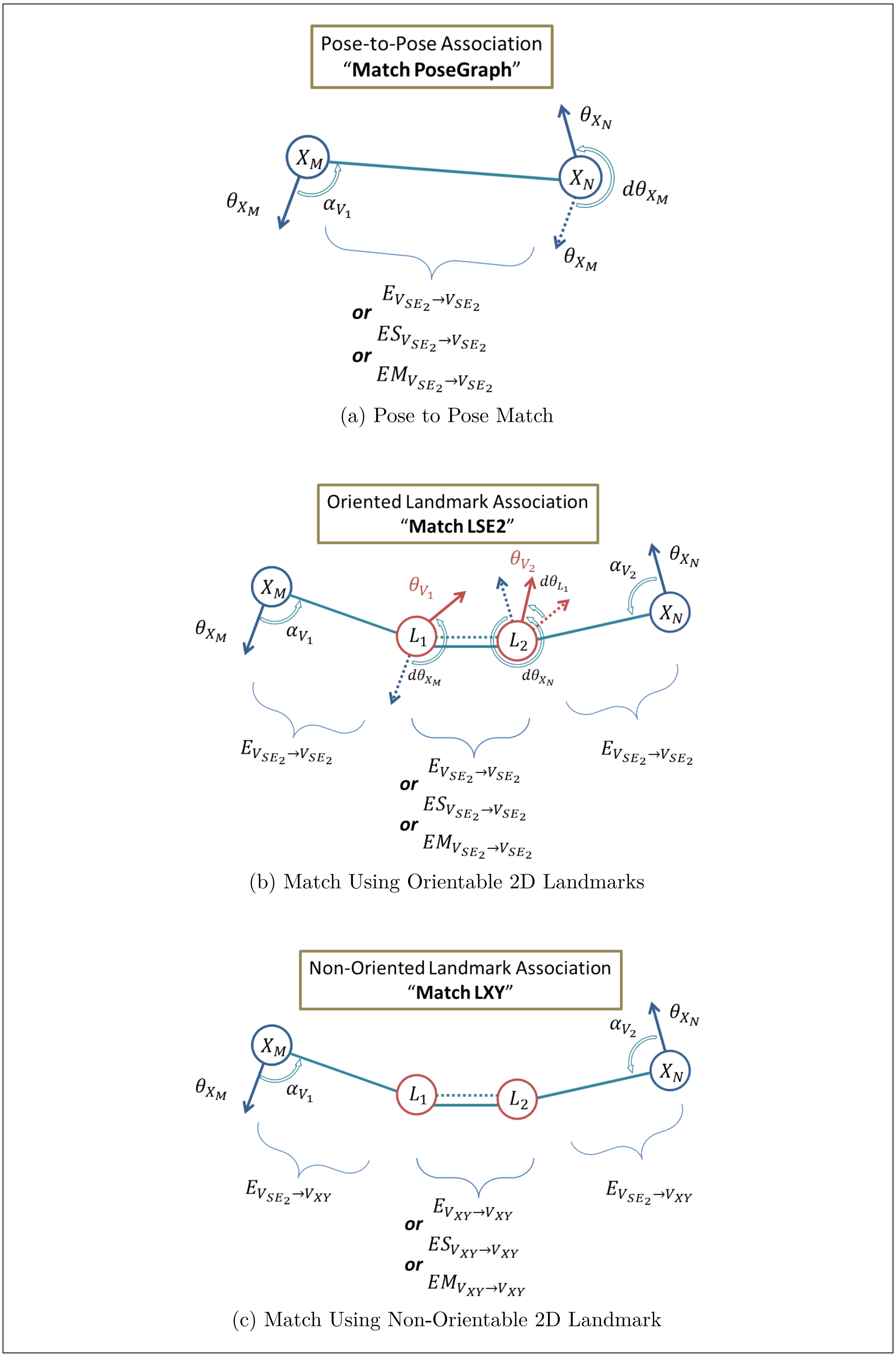
(a) Edge connecting a SE2 Vertex to a XY Vertex

(b) Edge connecting two XY Vertices

Figure 5.3: Edges Involving XY Vertices Used in The Graph.

The Swicthable constraint introduces an additional vertex that encodes the weight of the edge as described in the previous chapter. The initial weight and covariance matrix associated to the gap between the current state of the weight and its prior values are set by the front-end (either based on the user's choice or some data-association driven reasons). The MaxMixtures ones use two covariance matrices: one related to the observation model and another one with massive covariance terms that model the null hypothesis. The optimizer performs the hypotheses test at each iteration and apply the relevant matrix.

These are just elementary constraints. In order to encode data association, we created groups of vertices and edges that are specially designed to fit the across-track mode of observation of an AUV. They are described in Figure 5.4. The objective of these blocks is to model different types of associations based on the nature of the data that is being considered. Indeed, as discussed in the Chapter about sonar image processing, depending of the image processing algorithm, it is possible to extract either flat, possibly orientable features, or others like echo-shadows that are characterized by a well-defined measurement of the Slant Range. The rational behind these different types of matches is to transpose our methodology to an heterogeneous dataset in the future by clearly separating groups of data associations. These different matches are declined into their robust versions by inserting the correct elementary constraint at the appropriate place.


Pose-to-Pose Association
"Match PoseGraph"
θ_{X_N}
X_M
X_N
$d\theta_{X_M}$
θ_{X_M}
α_{V_1}
θ_{X_M}
$E_{V_{SE_2}\rightarrow V_{SE_2}}$
or
$ES_{V_{SE_2}\rightarrow V_{SE_2}}$
or
$EM_{V_{SE_2}\rightarrow V_{SE_2}}$
Oriented Landmark Association
"Match LSE2"
θ_{V_2}
$d\theta_{L_1}$
α_{V_2}
θ_{X_N}
θ_{V_1}
X_M
θ_{X_M}
α_{V_1}
L_1
L_2
X_N
$d\theta_{X_M}$
$d\theta_{X_N}$
$E_{V_{SE_2}\rightarrow V_{SE_2}}$
$E_{V_{SE_2}\rightarrow V_{SE_2}}$
$E_{V_{SE_2}\rightarrow V_{SE_2}}$
or
$ES_{V_{SE_2}\rightarrow V_{SE_2}}$
or
$EM_{V_{SE_2}\rightarrow V_{SE_2}}$
Non-Oriented Landmark Association
"Match LXY"
θ_{X_N}
α_{V_2}
X_M
X_N
θ_{X_M}
α_{V_1}
L_1
L_2
$E_{V_{SE_2}\rightarrow V_{XY}}$
$E_{V_{SE_2}\rightarrow V_{XY}}$
$E_{V_{XY}\rightarrow V_{XY}}$
or
$ES_{V_{XY}\rightarrow V_{XY}}$
or
$EM_{V_{XY}\rightarrow V_{XY}}$

(a) Pose to Pose Match

(b) Match Using Orientable 2D Landmarks

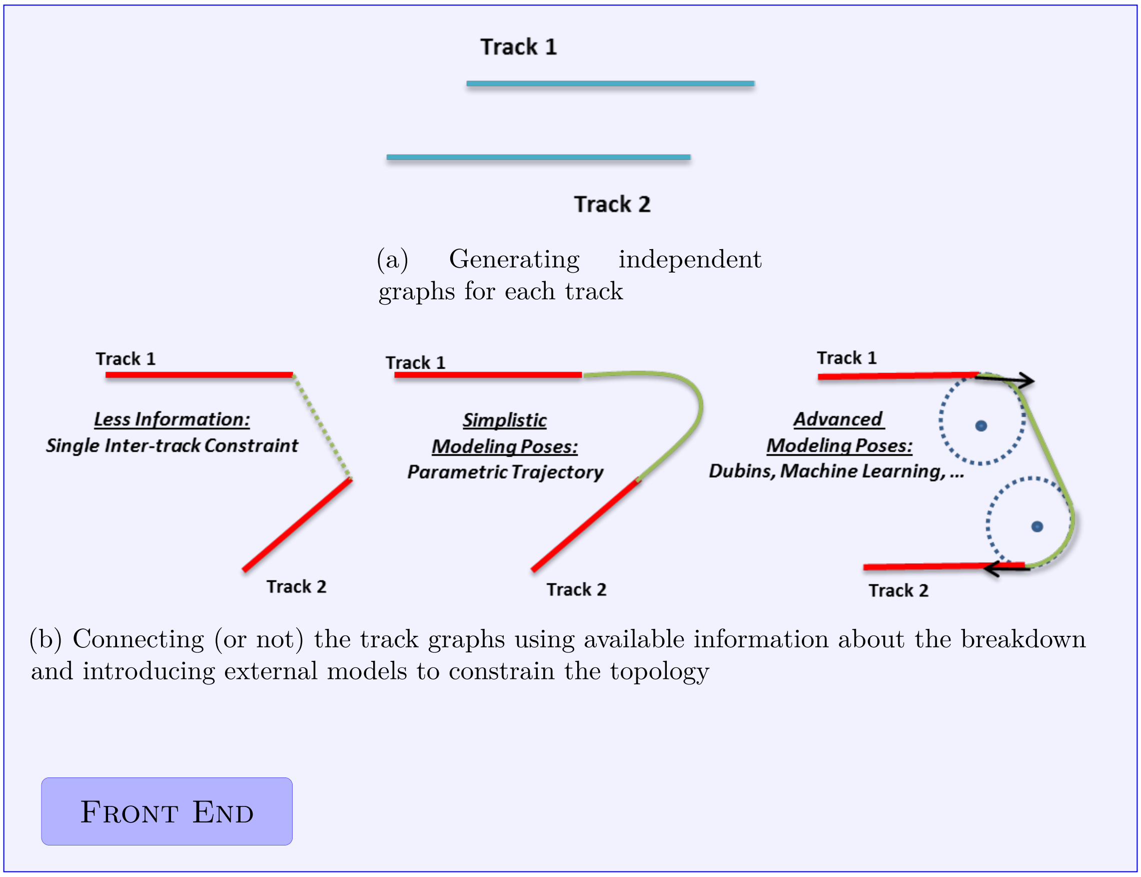
(c) Match Using Non-Orientable 2D Landmark

Figure 5.4: Classes of Matches That Have been Implemented in the Framework

5.2.3 Front-End: Building The Graph Backbone

As presented in the previous Chapter, the first step in building the graph consists in generating the motion constraint backbone. We developed the framework in such a way that the inter-track region that is modelling the AUV momentary breakdown in our scenario could be updated at will. Our algorithm works as a large database where all vertices and edges are associated with a series of attribute that facilitates to create different regions in the graph. After the optimization, it is possible to retrieve the actual state of the extremities of the "breakdown region" and remove, replace or change their parametrization based on criteria that can defined at the beginning of the scenario. This adaptability is actually critical for being able to impact the iterative gain of knowledge over the blackout area thanks to the sonar observations. Figure 5.5 presents several options below.

This is an interesting place in our framework where one can insert a machine learning based algorithm that can take a few parameters about the breakdown (duration, compass data etc...) in order to generate a modeling inter-track trajectory that enable a better first guess for the graph backbone.



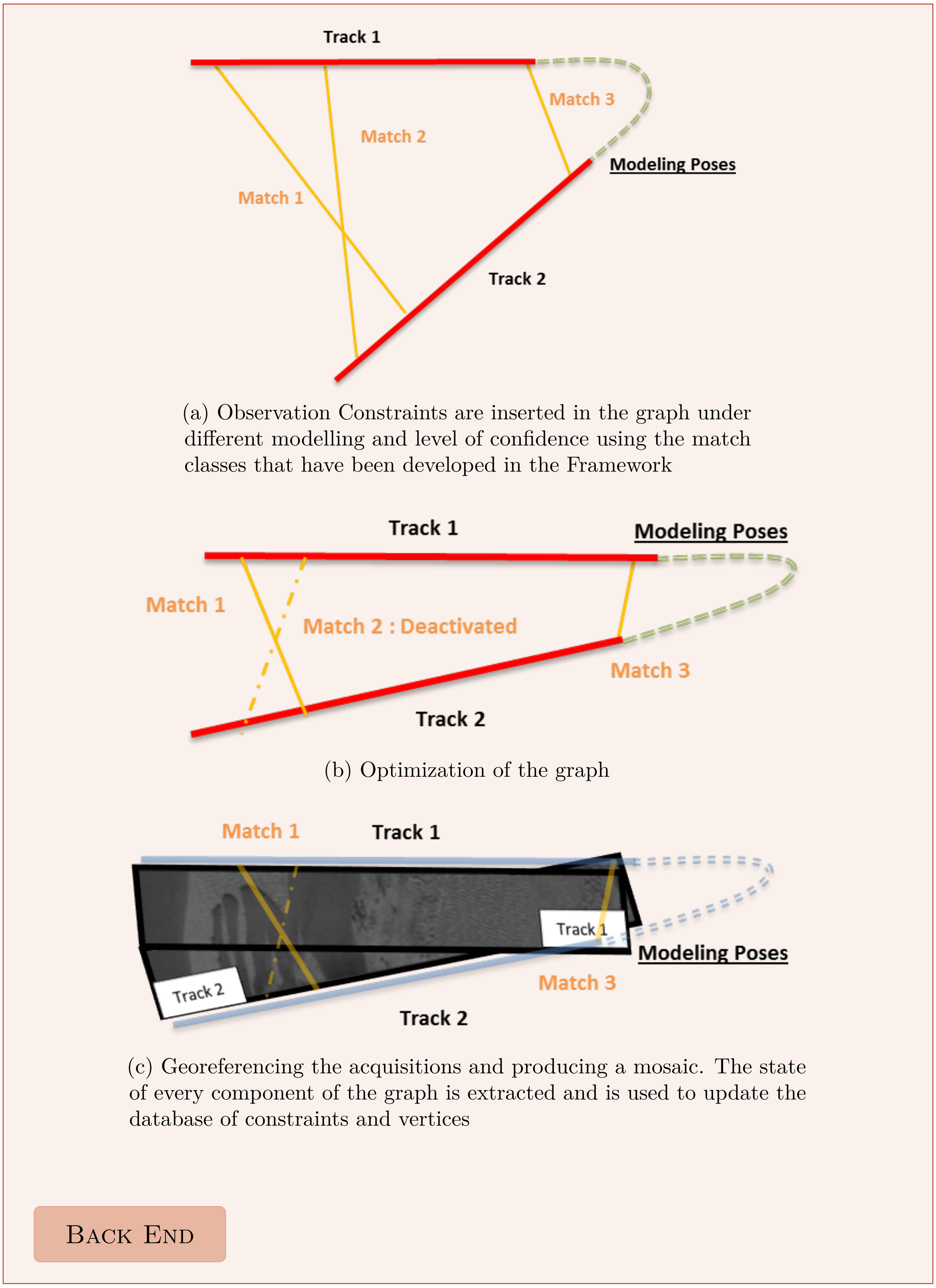
(a) Generating independent graphs for each track

(b) Connecting (or not) the track graphs using available information about the breakdown and introducing external models to constrain the topology

Figure 5.5: Building the Graph Backbone Using Motion Constraints derived from the INS and the Breakdown Hypotheses

5.2.4 Back-End: LCCs, Optimzation & Mosaicking

Figure 5.6 shows the principle steps of the back-end part.



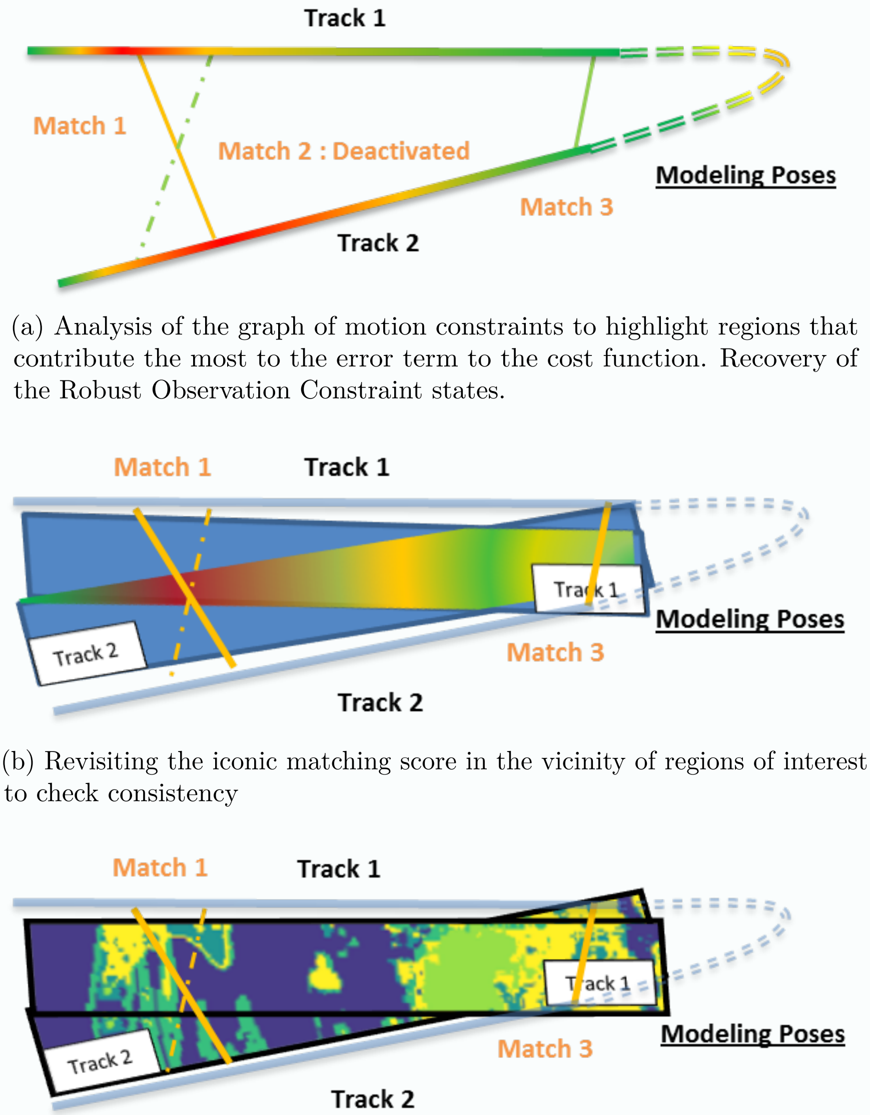
(a) Observation Constraints are inserted in the graph under different modelling and level of confidence using the match classes that have been developed in the Framework

(b) Optimization of the graph

(c) Georeferencing the acquisitions and producing a mosaic. The state of every component of the graph is extracted and is used to update the database of constraints and vertices

Figure 5.6: Inserting LCCs, Optimizing & Georeferencing Sonar Acquisitions

5.2.5 Post-Mosaicking Analysis



(a) Analysis of the graph of motion constraints to highlight regions that contribute the most to the error term to the cost function. Recovery of the Robust Observation Constraint states.

(b) Revisiting the iconic matching score in the vicinity of regions of interest to check consistency

(c) Comparing Texture classes from different overlapping regions to check consistency

Post Mosaicking Analysis

Figure 5.7: Combining different sources of information to measure the plausibility of the post-optimization topology

The Post-Mosaicking Analysis phase 5.7 is designed to study the plausibility of the mosaic generated by the current graph topology. The objective of our framework is to enable fusion of different sources of information, while being able to guide the search in specific regions. In fact, as highlighted in the chapter about sonar image processing, since every algorithm tends to have its strengths and flaws it seems particularly useful to focus some types of processing in regions where they are reliable. The fact that we use a database to encode the graph is very helpful at extracting vicinity of group of poses. This allows an in-depth analysis in the search of potential inconsistencies.

This part of the framework is very interesting for studying the coupling of fusion of heterogeneous metrics and also the relative efficiency of each independent analysis method using the ground truth as a baseline. In the next section, we present the outcome of the framework for different scenarios.

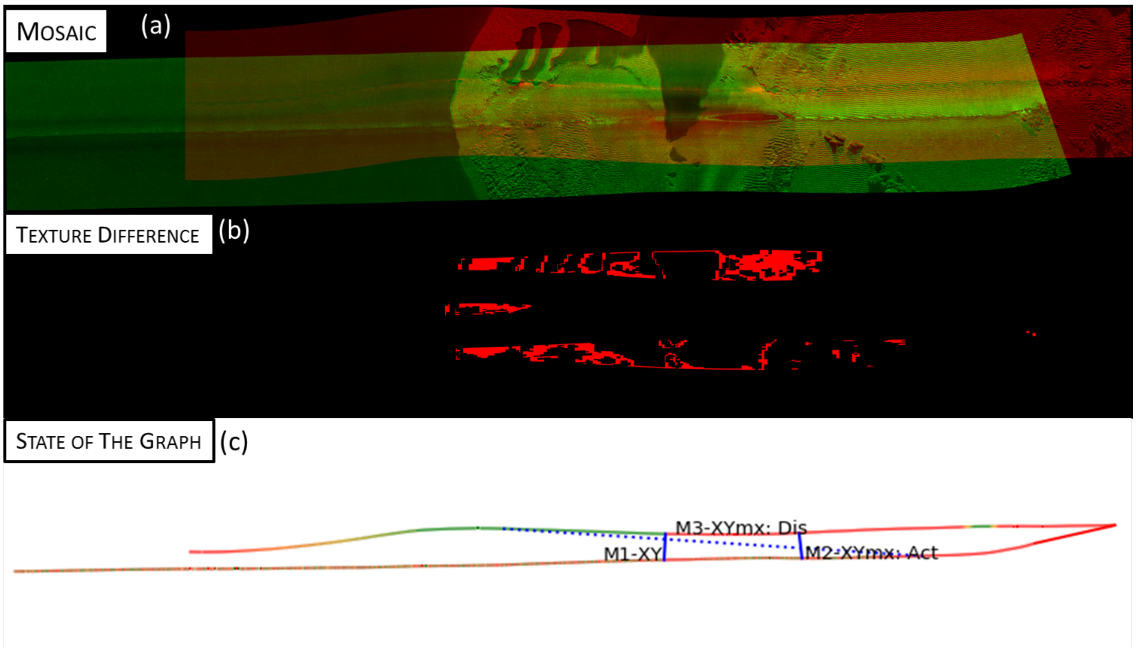


Figure 5.8: Example of an output of the framework using 3 manually extracted matches: two correct ones and one outlier:
(a) Georeferenced Mosaic
(b) Map representing the difference of texture classification between the two registered tracks
(c) Post Optimization Graph.
In red, regions that contributes to the most important part to the error term In green, regions that produce little to no error. The color scale is dynamically adjusted between min and max values, here most of the graph constraints generate the same amount of error, therefore a lot of red. The blue constraints represent the matches (LCCs), here Match 3, encoded as a MaxMix XY observation constraint, has been rejected by the optimizer, therefore the dotted line plotted by our Framework.

5.3 Generating a multi-layer mosaic

An example of the outcome of the working framework is presented in Figure5.8. A multi-layers georeferenced product is generated using the post-optimization graph state that is contained in the framework database. In plot (c) of Figure 5.8 the graph of the error terms includes the states and the reference of each matches as described in the previous section. The change of state can be further investigated by computing statistics in the vicinity of the poses involved in each observation association. While this first example shows a well distributed error term over the full trajectory (as the color are set via a dynamic range between the max and the min), the following example is a lot more characteristic to what happen to the graph topology is severely impacted by a group of LCCs.

Indeed, as shown in Figure 5.9 , most of the error cost is located in a specific neighbourhood. This is a relevant piece of information in order to guide the search for in-depth validation techniques that would need to project the mosaic. Indeed, from an operational point of view, re-projecting the sonar acquisitions each time the trajectory is impacted represents a significant cost in terms of resources and time. This may be a problem if the AUV is in a dangerous environment where being able to quickly take the good decision is paramount. Consequently directly focus around such region of interest may be a time-efficient option. One interesting strategy in the context of heterogeneous dataset is to study the other type of modality that was not used to form the Observation Constraint in the vicinity. This may constitute a way to validate the scene registration using a complementary source of information.

The example presented in Figure 5.10 highlights the importance of the turn parameterization in the context of numerous outliers (that have deactivated). When looking at the detail extracted from the previous plot presented in Figure 5.11, it is possible to spot a semi-activated switchable constraint that reflects the trade-off made by the optimizer in a context of numerous possible topologies. Once again, this is an indicator that can raise suspicions and may trigger some additional checks.

By limiting the mosaic validation by directly targeting points of interest, it is possible to afford to run several topology tests as computing the optimized graph of constraints is significantly less time expansive than generating large blocs of georeferenced and multi-layers mosaics.

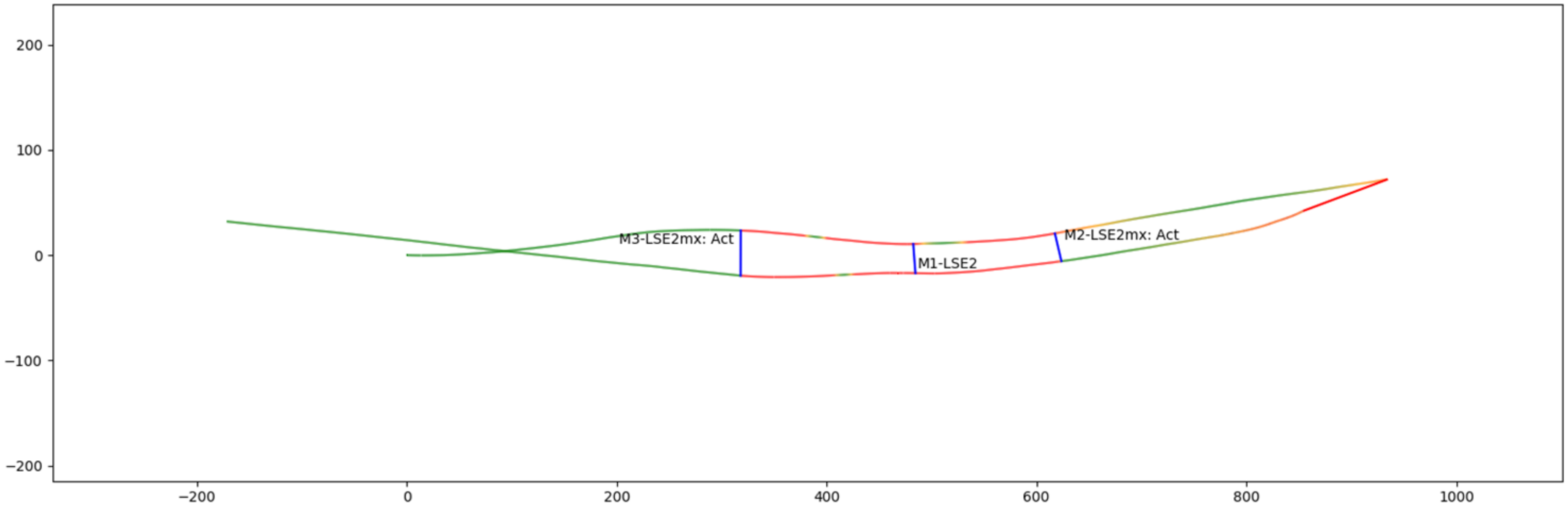


Figure 5.9: Example of Graph where LLCs effect is concentrated

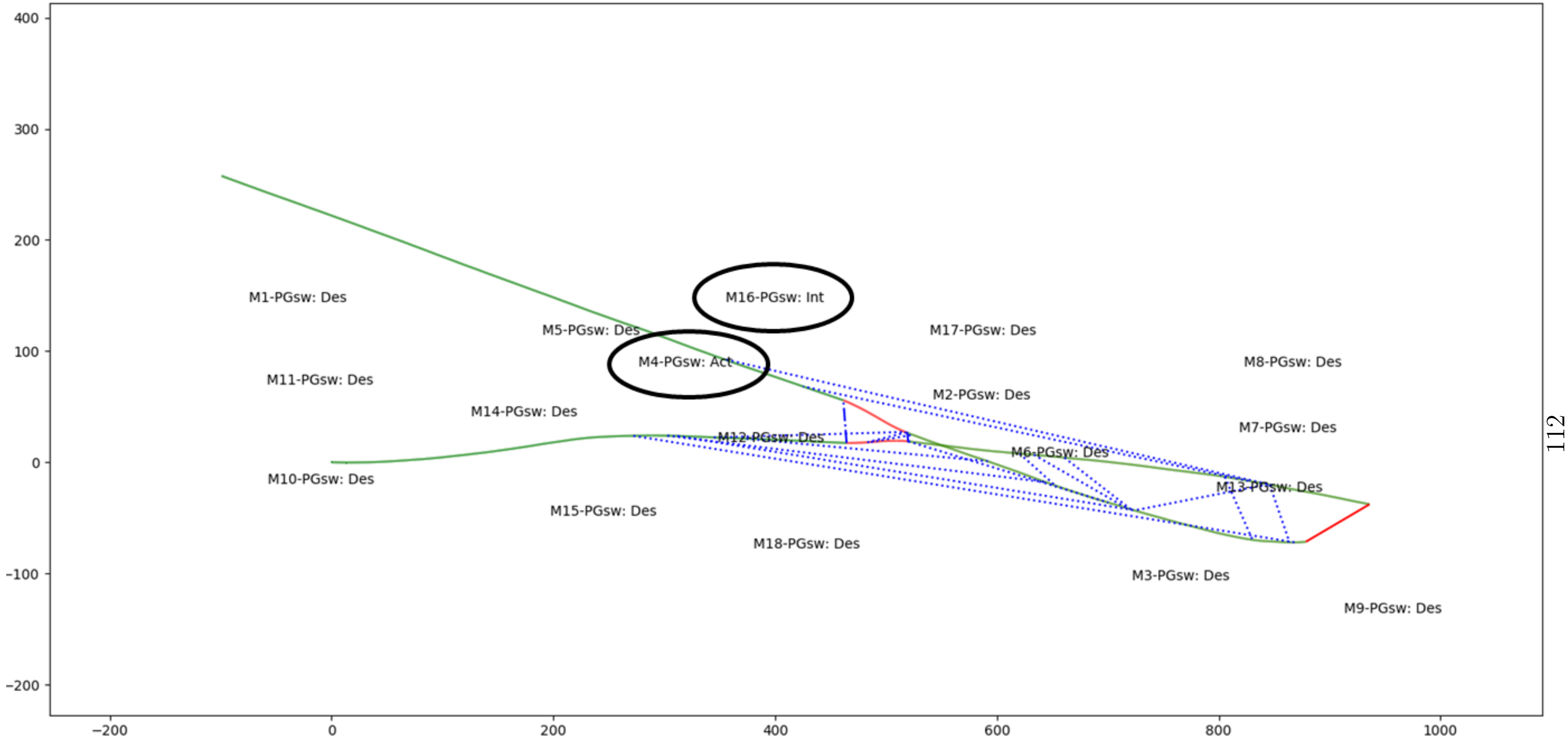


Figure 5.10: Example of Graph using a set of numerous outliers & encoding LCCs with switchable constraints. While most of the inconsistent constraints are deactivated, we can see that two regions of the graph clearly support most of the error term generated by the graph. In the red region in the middle, one switchable constraint is also in an intermediary state.

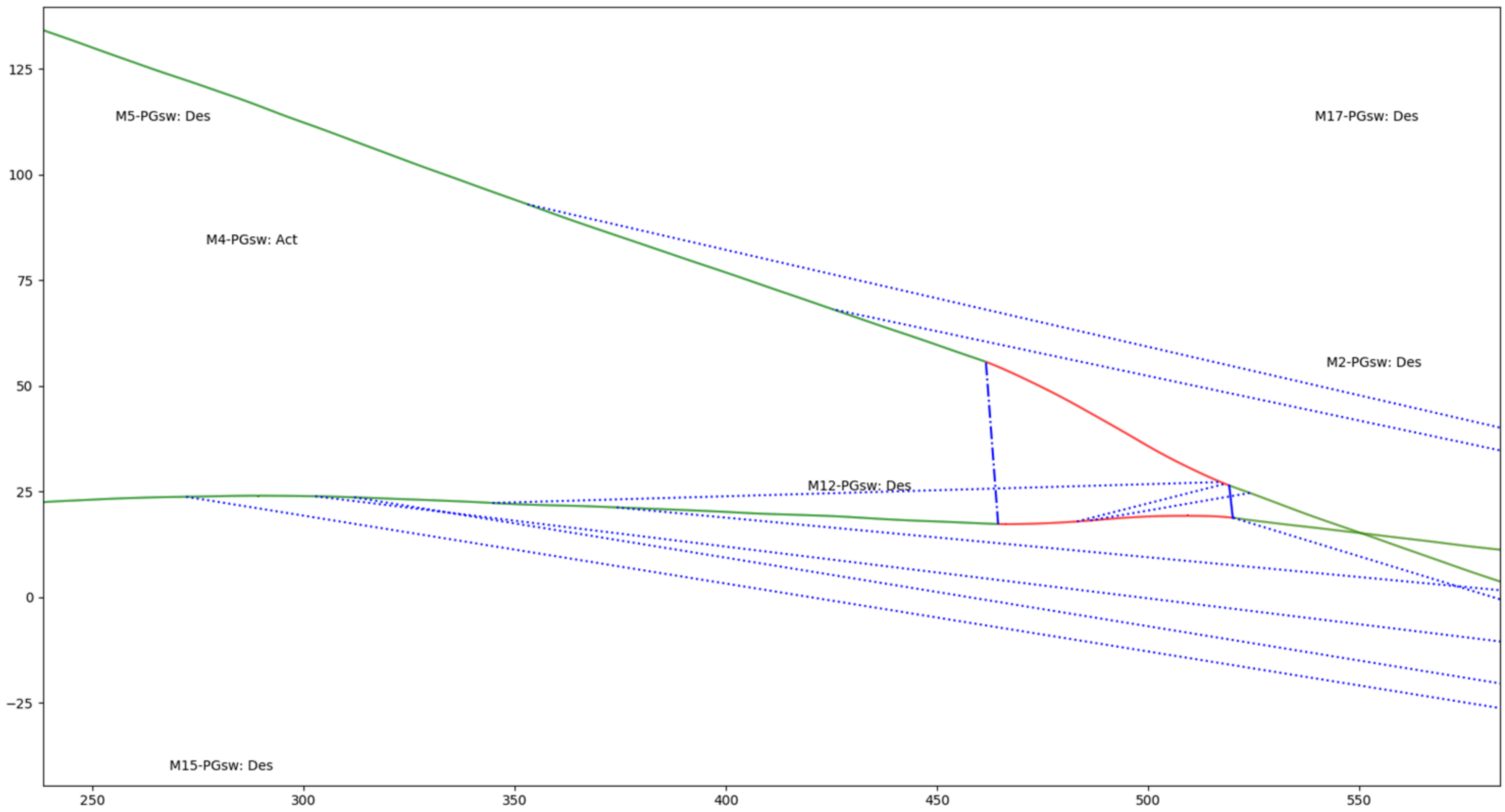


Figure 5.11: Focus on the vicinity of the activated observation constraints of the graph presented in 5.10
The line/dot edge is the partially activated switch, corresponding to Match 4 encoded as a Switchable Pose Graph Constraint.

5.4 Perspectives

The Robust SLAM framework, that has been developed during our research, offers several interesting perspectives in terms of heterogeneous data fusion. Among its features, the ability to revisit the graph topology, both in terms of LCCs, but also by introducing modelling poses during the operational breakdown offers key research opportunities.

In terms of navigation-related applications:

- Assessing how the quality of the INS impacts the search of the correct topology with a more accurate motion modelling. An interesting application would be understanding how far the SLAM algorithm can complement a low-grade, but affordable, INS. One application would be to downsize the INS requirements to lower the overall cost of the AUV.
- Measuring the effect of the deactivation/alteration of a single motion measurement sensor in order to model the effect an incapacitated instrument. This could model the effects that may result from operational damages.
- Develop and test a Machine-Learning based algorithm to further constrain the search for the correct topology by using a very limited amount of inputs parameters to define a robust inter-track modeling.

In terms of underwater remote sensing applications:

- Use the topology analysis to guide multi-modal "patch" matching strategies, in order to complement the use of different remote sensing instruments.
- Benchmark state-of-the-art sonar data association techniques to compare their respective efficiency in a global registration problem. Analysis on different Non Linear Scale Space modelling to improve global landmark search.
- Identify robust fusion scores that are based on the strengths and weaknesses of different ways to process sonar data.

Chapter 6

Discussions, Conclusion & Future Work

In this study we tackle an original and complex research problem in the field of naval robotics: how to relocate an AUV that got lost after a breakdown without surfacing in a hostile environment?

Discussions And Conclusion

In order to tackle this demanding operational scenario, we used a strategy based on graph SLAM algorithms facing a global registration problem.

Consequently, our research work focuses on three different axes:

- The detection and association of salient points of the underwater environment using side scan sonar images in a global registration context. A particular focus has been given to improving the detection of natural landmarks by targeting regions where the associated matching methods are appropriate.
- The study of the behavior and efficiency of robust graph-SLAM algorithms, in particular when facing two specific kind of operational complications. The first one comes from the rupture of the motion measurement "backbone" which leaves the AUV in a situation similar to the *Stolen Robot Problem*. The other challenge arises from the coupling of the unstructured nature of the underwater environment with the specificities of an active lateral sensor. This unfortunate combination may lead to poor data association and therefore erroneous loop-closure constraints.
- The development of a novel SLAM framework that allows to iteratively generate a graph with an evolving topology that can be improved by detecting and removing ambiguities arising from data association errors and trajectory interruptions.

With regards to sonar imagery, we aimed to overcome the limitations of traditional techniques that usually focus on local registration problems. In particular, we were interested in ways to make patch-matching methods more suitable for the construction of Loop closure constraints, necessary for the creation of the SLAM graph. We were able to show that such techniques are mainly relevant in regions of the seafloor that exhibit relative flatness and are therefore less vulnerable to viewpoint change and grazing angle variation. We also noted that the use of certain edge-preserving filters makes the patch-matching method more selective for the association phase.

We have also addressed the problem of the excessively exhaustive nature of such method by conducting two complementary studies. One of them consists in segmenting the textural information of the sonar mosaic in order to limit the detection to regions that react particularly well to the similarity measure used, here MICR. The other idea was to promote efficient detection of macroscopic features by drawing inspiration from the methods implemented in keypoints extractor as used in classical robotics. We explored the potential of different image scale spaces (linear or not) to preserve the global structure of the scene.

We also concluded that a series of treatments (filtering, contrast improvement and texture classification) made it possible to observe the repeatability of certain keypoints among different tracks, a result that is usually complex to obtain with sonar images. Our work presents interesting observations that would be helpful for designing a keypoint detector tailored for active lateral sensors, either sonar or radar. However, further research would be still required and a special care should be given to the variation of the SNR ratio and the sensor resolution over the range.

Furthermore, due to the low structure of the seafloor, we noted that it was critical to adopt a data association strategy that was combining independent sonar data processing methods in parallel. In particular, constraining the use of specific image processing algorithms to the type of regions where they perform better should be part of the foundation of an adaptable and robust framework. For instance, the variation of the sea bottom elevation could be used to actually inform the side scan sonar image processing part. Indeed, locally plane regions tend to react quite differently to iconic methods that the ones presenting some local variation (ripples, echo-shadow complex etc...) as we observed in this study. Conversely, algorithms specialized in detecting elevated landmarks such as echo-shadows should be applied in priority to regions were the seafloor elevation varies a lot more. A prior fast scene segmentation could therefore generate regions in the mosaic where only specific detection algorithms would be applied to therefore reduce the amount of computations. This would pave the way to heterogeneous dataset processing (image and bathymetry) and would provide the SLAM algorithm with different families of matches (2D for flat regions and 3D for elevated features).

After concluding that a graph modelling of our map and trajectory estimation problem was the preferred strategy, we investigated the effects of the vehicle temporary breakdown over robust optimization-based SLAM algorithms. We noted how the interruption of the “motion constraints backbone” turned the search for the correct scene topology into a delicate problem. Rather than trying to relocate the vehicle by modelling the problem as the association of two independent tracks, we suggest a methodology that aims to use the scarce available information to constrain the “breakdown region” as much as possible. Ideally, the level of knowledge of the system about that region is supposed to increase when finding correct observation constraints. Refining the parameterization of the region of the graph, while keeping it flexible, is key to the recovery of the true scene topology following the temporary breakdown.

We also ran experiments to measure how Loop-Closure Constraints play a part in their vicinity. We tested variation of observation constraints (switchable or maxmixtures in particular) to get some insights about their stregths and limits in our specific scenario. We concluded that in some cases, the addition of poor quality data association with an int erupted trajectory "backbone"could eventually lead to erroneous LCCs prevailing over correct ones during the optimization. The key to overcome that issue is to perform analysis of the post-SLAM mosaic to either detect "good" overlaps and reiforce the part played by the LCCs in their vicinity, or conversely, discard LCCs surrounded by an unlikely mosaic.

Consequently, we developed a novel SLAM framework that is able to encode diverse ways to associate data while keeping these associations compatible with some robust graph constraints presented in the literature. These types of associations are derived from the intuition that two types of remarkable feature of the underwater environment should give enough information to address the high level of uncertainty due to the breakdown. Our Framework enables the reconfiguration of the graph topology between successive runs in order to iteratively refine the scene structure. For instance, it is possible to easily introduce modelling motion constraints in the uncertainty region in an orderly way. It is therefore possible to improve the parametrization by adding/removing poses or updating modelling motion constraints to reflect the gain of information thanks to the image processing part. Tools to analyze the consistency of the *a posteriori* mosaic were also implemented. In particular, we monitor the error of each constraint to detect regions of “tensions” in the graph, using a mass-spring analogy. Texture matching is used to check the overall symbolic consistency of the scene and iconic similarity measures can also be applied to regions where they perform well to measure the correlations of overlapping patches. Our framework relies on a database model and a search can be executed in very specific regions to further investigate the effect of the graph optimization.

There are several axes of improvement for our proposed framework. First, the current motion model can be complexified to more accurately reflect the dynamics of the AUV. Same for the pre-processing phase of the sonar signal. Using the actual antenna parameters coupled with a more accurate 3D pose modelling may dramatically improve the intensity correction. Furthermore, this tool can be improved by inserting more advanced data association algorithm at the front-end level. Our objective in this study was to actually gain insights about the strength and limitation of the usual techniques in a harsh and challenging scenario to inform the design of a robust SLAM-centered framework. Actually, each specific topic addressed in this thesis requires a unique and significant level of expertise and has its own community of researchers.

Future Work

Overall, our research raises a list of interesting future perspectives to go further in the improvement of underwater robotics robustness:

- First, developing an active sensor keypoint detector seems to be a very relevant field of study. The use of Non Linear Scale Space should be further investigated with a particular attention to the families of filters that are particularly efficient at removing the speckle noise. This could also have application to the field of active radar sensors.
- Second, there is an interesting insertion point in the framework for the use of a machine learning algorithm that could be used to generate possible trajectories within the breakdown region. It could be seen as a black box that uses a few parameters in entry (duration of breakdown, water currents, compass information, etc...) coupled with a supervised algorithm that mapped a relation between these parameters and likely trajectories. This could be used to recover some precious information lost during the breakdown. Basically it is like trying to improve the resolution of an image of an apple: even if we have little pixels to describe the apple, we can use the knowledge learned through thousands of apple images to suggest a likely high resolution representation of the object.
- Third, the a posteriori mosaic analysis can be further developed by identifying efficient score function that perform well using very different type of information like textures, patches intensity or the local uncertainty about the trajectory.
- Last but not least, heterogeneous sensor data processing could be studied in such framework. For instance if available bathymetric data was available, it could dramatically increase the data association processing and help in recovering the topology faster.

Overall, underwater robotics offers by essence a very challenging environment that pushes many current innovations and techniques to their limits. However, with the exponential growth on in board computational capabilities and the current thrive of Artificial Intelligence, there is no doubt that AUVs will be able to tackle more and more challenging missions while processing massive amount of data.

The supremacy of the seas was, is and will remain a strategic objective for many nations to preserve their interests. The decades to come will see the increase of navies' level of automation. The development of unrivaled AUVs will soon become as important as nuclear deterrence itself.

References

[1] Klein system 5000 v2 side scan sonar - datasheet. `http://www.kleinmarinesystems.com/PDF/datasheets/Klein_System_5000_AUV-V2_rev0718.pdf`. Accessed: 2021-06-20.

[2] Jiaqiu Ai, Ruitian Tian, Qiwu Luo, Jing Jin, and Bo Tang. Multi-scale rotation-invariant haar-like feature integrated cnn-based ship detection algorithm of multiple-target environment in sar imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12):10070–10087, 2019.

[3] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. Kaze features. In *European conference on computer vision*, pages 214–227. Springer, 2012.

[4] Alex Alcocer, Paulo Oliveira, and Antonio Pascoal. Underwater acoustic positioning systems based on buoys with gps. In *Proceedings of the Eighth European Conference on Underwater Acoustics*, volume 8, pages 1–8, 2006.

[5] Rosario Aragues, Eduardo Montijano, and Carlos Sagues. Consistent data association in multi-robot systems with limited communications. In *Robotics: Science and Systems*, pages 97–104, 2011.

[6] Josep Aulinas, Amir Fazlollahi, Joaquim Salvi, Xavier Lladó, Y Petillot, Jamil Sawas, and Rafael Garcıa. Robust automatic landmark detection for underwater slam using side-scan sonar imaging. In *Proceedings of the 11th International Conference on Mobile Robots and Competitions*, pages 21–26, 2011.

[7] P. Baccou, B. Jouvencel, V. Creuze, and C. Rabaud. Cooperative positioning and navigation for multiple auv operations. In *MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No.01CH37295)*, volume 3, pages 1816–1821 vol.3, Nov 2001.

[8] Philippe Baccou and Bruno Jouvencel. Homing and navigation using one transponder for auv, postprocessing comparisons results with long base-line navigation. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 4, pages 4004–4009. IEEE, 2002.

[9] Alexander Bahr, John J Leonard, and Maurice F Fallon. Cooperative localization for autonomous underwater vehicles. *The International Journal of Robotics Research*, 28(6):714–728, 2009.

[10] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam): Part ii. *IEEE robotics & automation magazine*, 13(3):108–117, 2006.

[11] Stephen Barkby, Stefan B Williams, Oscar Pizarro, and Michael V Jakuba. Bathymetric particle filter slam using trajectory maps. *The International journal of robotics research*, 31(12):1409–1430, 2012.

[12] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[13] Loïc Bernicola, Didier Gueriot, and Jean-Marc Le Caillec. A hybrid registration approach combining slam and elastic matching for automatic side-scan sonar mosaic. In *2014 Oceans-St. John's*, pages 1–5. IEEE, 2014.

[14] Tim Berthold, Artem Leichter, Bodo Rosenhahn, Volker Berkhahn, and Jennifer Valerius. Seabed sediment classification of side-scan sonar data using convolutional neural networks. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8. IEEE, 2017.

[15] IEEE-SA Standards Board. IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fiber Optic Gyros, Annex C. *IEEE STD 952-1997(R2008)*, 2008.

[16] Ned A Brokloff. Matrix algorithm for doppler sonar navigation. In *Proceedings of OCEANS'94*, volume 3, pages III–378. IEEE, 1994.

[17] Antoni Burguera and Gabriel Oliver. Intensity correction of side-scan sonar images. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, pages 1–4. IEEE, 2014.

[18] Antoni Burguera and Gabriel Oliver. High-resolution underwater mapping using side-scan sonar. *PloS one*, 11(1):e0146396, 2016.

[19] I. Rouse C. Flewellen. The acoustic navigation of a deep towed underwater vehicle using a reversed ultra short baseline method. In *Proceedings of the Seventh European Conference on Underwater Acoustics, Delft, Netherlands*. ECUA 2004, 5-8 July 2004.

[20] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

[21] CG Capus, AC Banks, E Coiras, I Tena Ruiz, CJ Smith, and YR Petillot. Data correction for visualisation and classification of sidescan sonar imagery. *IET radar, sonar & navigation*, 2(3):155–169, 2008.

[22] Turgay Celik and Tardi Tjahjadi. A novel method for sidescan sonar image segmentation. *IEEE Journal of Oceanic Engineering*, 36(2):186–194, 2011.

[23] Cyril Chailloux. *Recalage d'images sonar par appariemment de régions: application à la génération d'une mosaïque.* PhD thesis, Télécom Bretagne, 2007.

[24] Cyril Chailloux, Jean-Marc Le Caillec, Didier Gueriot, and Benoit Zerr. Intensity-based block matching algorithm for mosaicing sonar images. *IEEE Journal of Oceanic Engineering*, 36(4):627–645, 2011.

[25] Robert D. Christ and Robert L. Wernli Sr. Chapter 4 - underwater acoustics and positioning. In Robert D. Christ and Robert L. Wernli Sr, editors, *The ROV Manual*, pages 81 – 124. Butterworth-Heinemann, Oxford, 2007.

[26] Brian Claus, James H. Kepper IV, Stefano Suman, and James C. Kinsey. Closed-loop one-way-travel-time navigation using low-grade odometry for autonomous underwater vehicles. *Journal of Field Robotics*, 35(4):421–434, 2018.

[27] Sylvie Daniel, Fabrice Le Leannec, Christian Roux, B Soliman, and Eric P Maillard. Side-scan sonar image matching. *IEEE Journal of Oceanic Engineering*, 23(3):245–259, 1998.

[28] Jean Dezert and Yaakov Bar-Shalom. Joint probabilistic data association for autonomous navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 29(4):1275–1286, 1993.

[29] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[30] Armagan Elibol, Nuno Gracias, and Rafael Garcia. Augmented state–extended kalman filter combined framework for topology estimation in large-area underwater mapping. *Journal of Field Robotics*, 27(5):656–674, 2010.

[31] R. M. Eustice, L. L. Whitcomb, H. Singh, and M. Grund. Recent advances in synchronous-clock one-way-travel-time acoustic navigation. In *OCEANS 2006*, pages 1–6, Sep. 2006.

[32] Michael Felsberg and Gerald Sommer. The monogenic signal. *IEEE transactions on signal processing*, 49(12):3136–3144, 2001.

[33] Kenneth Gade. The seven ways to find heading. *The Journal of Navigation*, 69(5):955–970, 2016.

[34] Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, volume 140, pages 107–113. IET, 1993.

[35] Giorgio Grisetti, Rainer Kümmerle, Cyrill Stachniss, and Wolfram Burgard. A tutorial on graph-based slam. *IEEE Intelligent Transportation Systems Magazine*, 2(4):31–43, 2010.

[36] Giorgio Grisetti, Rainer Kümmerle, Hauke Strasdat, and Kurt Konolige. g2o: A general framework for (hyper) graph optimization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China*, pages 9–13, 2011.

[37] Paul D. Groves. *Principles of GNSS, Inertial and Multisensor Integrated Systems, 2nd Edition.* Artech House, 2013.

[38] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and P-J Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002.

[39] Per Espen Hagen, Oivind Midtgaard, and Oistein Hasvold. Making auvs truly autonomous. In *OCEANS 2007*, pages 1–4. IEEE, 2007.

[40] Robert M Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[41] Peter J. Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73 – 101, 1964.

[42] Mahamadou Idrissa and Marc Acheroy. Texture classification using gabor filters. *Pattern Recognition Letters*, 23(9):1095–1102, 2002.

[43] Mathieu Issartel, Didier Guériot, Nabil Aouf, and Jean-Marc Le Caillec. Robust slam for side scan sonar image mosaicking. In *OCEANS 2017-Anchorage*, pages 1–10. IEEE, 2017.

[44] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998.

[45] H Paul Johnson and Maryann Helferty. The geological interpretation of side-scan sonar. *Reviews of Geophysics*, 28(4):357–380, 1990.

[46] Simon J Julier and Jeffrey K Uhlmann. New extension of the kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. International Society for Optics and Photonics, 1997.

[47] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert. isam2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, 31(2):216–235, 2012.

[48] Michael Kaess, Ananth Ranganathan, and Frank Dellaert. isam: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, 2008.

[49] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[50] Behzad Kamgar-Parsi and Behrooz Kamgar-Parsi. Vehicle localization on gravity maps. In *Unmanned Ground Vehicle Technology*, volume 3693, pages 182–191. International Society for Optics and Photonics, 1999.

[51] Imen Karoui, Ronan Fablet, Jean-Marc Boucher, and Jean-Marie Augustin. Seabed segmentation using optimized statistics of sonar textures. *IEEE Transactions on geoscience and remote sensing*, 47(6):1621–1631, 2009.

[52] A Kenny. Recent Advances In Doppler Sonar Technology For Subsea Navigation And Current Profiling. In *Oceanology International 2013, China*, 2013.

[53] Been Kim, Michael Kaess, Luke Fletcher, John Leonard, Abraham Bachrach, Nicholas Roy, and Seth Teller. Multiple relative pose graphs for robust cooperative mapping. In *2010 IEEE International Conference on Robotics and Automation*, pages 3185–3192. IEEE, 2010.

[54] Peter King, Benjamin Anstey, and Andrew Vardy. Comparison of feature detection techniques for auv navigation along a trained route. In *2013 OCEANS-San Diego*, pages 1–8. IEEE, 2013.

[55] Peter King, Andrew Vardy, Peter Vandrish, and Benjamin Anstey. Real-time side scan image generation and registration framework for auv route following. In *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, pages 1–6. IEEE, 2012.

[56] Lindsay Kleeman and Roman Kuc. Sonar sensing. In *Springer handbook of robotics*, pages 491–519. Springer, 2008.

[57] Naveen Kumar, Urbashi Mitra, and Shrikanth S Narayanan. Robust object classification in underwater sidescan sonar images by using reliability-aware fusion of shadow features. *IEEE Journal of oceanic engineering*, 40(3):592–606, 2014.

[58] Mikael Bliksted Larsen. Synthetic long baseline navigation of underwater vehicles. In *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No. 00CH37158)*, volume 3, pages 2043–2050. IEEE, 2000.

[59] Yasir Latif, César Cadena, and José Neira. Robust loop closing over time for pose graph slam. *The International Journal of Robotics Research*, 32(14):1611–1626, 2013.

[60] Isabelle Leblond, Michel Legris, and Basel Solaiman. Use of classification and segmentation of sidescan sonar images for long term registration. In *Europe Oceans 2005*, volume 1, pages 322–327. IEEE, 2005.

[61] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

[62] Maria Lianantonakis and Yvan R Petillot. Sidescan sonar segmentation using texture descriptors and active contours. *IEEE Journal of Oceanic Engineering*, 32(3):744–752, 2007.

[63] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[64] José Melo and Aníbal Matos. Survey on advances on terrain based navigation for autonomous underwater vehicles. *Ocean Engineering*, 139:250–264, 2017.

[65] Max Mignotte, Christophe Collet, Patrick Perez, and Patrick Bouthemy. Hybrid genetic optimization and statistical model based approach for the classification of shadow shapes in sonar imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(2):129–141, 2000.

[66] P-Y Mignotte, Maria Lianantonakis, and Yvan Petillot. Unsupervised registration of textured images: applications to side-scan sonar. In *Europe Oceans 2005*, volume 1, pages 622–627. IEEE, 2005.

[67] Michael Montemerlo and Sebastian Thrun. Simultaneous localization and mapping with unknown data association using fastslam. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 2, pages 1985–1991. IEEE, 2003.

[68] Ahmed Nait-Chabane, Benoit Zerr, and Gilles Le Chenadec. Sidescan sonar imagery segmentation with a combination of texture and spectral analysis. In *2013 MTS/IEEE OCEANS-Bergen*, pages 1–6. IEEE, 2013.

[69] Florian Nicolas, Andreas Arnold-Bos, Isabelle Quidu, and Benoit Zerr. Rigid sonar tracks registration for mcm survey missions. In *Undersea Defence Technology (UDT)*, 2016.

[70] F Novella, M Ponchart, P Benet, Bosser P, and B Clement. State-of-the-Art of Standalone Accurate AUV Positionning - Application to High Resolution Bathymetric Surveys. June 2019.

[71] Edwin Olson and Pratik Agarwal. Inference on networks of mixtures for robust robot mapping. *The International Journal of Robotics Research*, 32(7):826–840, 2013.

[72] Edwin Olson, John J Leonard, and Seth Teller. Robust range-only beacon localization. *IEEE Journal of Oceanic Engineering*, 31(4):949–958, 2006.

[73] International Hydrographic Organization. 44. standards for hydrographic surveys, 2008.

[74] Alberto Ortiz, Javier Antich, and Gabriel Oliver. A particle filter-based approach for tracking undersea narrow telecommunication cables. *Machine Vision and Applications*, 22(2):283–302, 2011.

[75] Liam Paull, Guoquan Huang, Mae Seto, and John J Leonard. Communication-constrained multi-auv cooperative slam. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 509–516. IEEE, 2015.

[76] Ana Petrovic, O Divorra Escoda, and Pierre Vandergheynst. Multiresolution segmentation of natural images: from linear to nonlinear scale-space representations. *IEEE Transactions on Image Processing*, 13(8):1104–1114, 2004.

[77] Max Pfingsthorn and Andreas Birk. Generalized graph slam: Solving local and global ambiguities through multimodal and hyperedge constraints. *The International Journal of Robotics Research*, 35(6):601–630, 2016.

[78] Max Pfingsthorn, Heiko Bülow, Andreas Birk, F Ferreira, G Veruggio, M Caccia, and G Bruzzone. Large-scale mosaicking with spectral registration based simultaneous localization and mapping (ifmi-slam) in the ligurian sea. In *2013 MTS/IEEE OCEANS-Bergen*, pages 1–6. IEEE, 2013.

[79] Minh Tân Pham and Didier Gueriot. Guided block-matching for sonar image registration using unsupervised kohonen neural networks. In *2013 OCEANS-San Diego*, pages 1–5. IEEE, 2013.

[80] Laurent Picard, Alexandre Baussard, Isabelle Quidu, and Gilles Le Chenadec. Seafloor description in sonar images using the monogenic signal and the intrinsic dimensionality. *IEEE Transactions on Geoscience and Remote Sensing*, 56(9):5572–5587, 2018.

[81] M. J. D. POWELL. A hybrid method for nonlinear equations. *Numerical Methods for Nonlinear Algebraic Equations*, 1970.

[82] Scott Reed, Yvan Petillot, and J Bell. Automated approach to classification of mine-like objects in sidescan sonar using highlight and shadow information. *IEE Proceedings-Radar, Sonar and Navigation*, 151(1):48–56, 2004.

[83] Scott Reed, Ioseba Tena Ruiz, Chris Capus, and Yvan Petillot. The fusion of large scale classified side-scan sonar image mosaics. *IEEE transactions on image processing*, 15(7):2049–2060, 2006.

[84] Donald Reid. An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control*, 24(6):843–854, 1979.

[85] David Ribas, Pere Ridao, Jose Neira, and Juan D Tardos. Slam using an imaging sonar for partially structured underwater environments. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5040–5045. IEEE, 2006.

[86] Paul Rigby, Oscar Pizarro, and Stefan B Williams. Towards geo-referenced auv navigation through fusion of usbl and dvl measurements. In *OCEANS 2006*, pages 1–6. IEEE, 2006.

[87] Alexis Roche, Grégoire Malandain, Xavier Pennec, and Nicholas Ayache. The correlation ratio as a new similarity measure for multimodal image registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 1115–1124. Springer, 1998.

[88] Stergios I Roumeliotis and George A Bekey. Distributed multirobot localization. *IEEE transactions on robotics and automation*, 18(5):781–795, 2002.

[89] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

[90] I Tena Ruiz, Sébastien De Raucourt, Yvan Petillot, and David M Lane. Concurrent mapping and localization using sidescan sonar. *IEEE Journal of Oceanic Engineering*, 29(2):442–456, 2004.

[91] I Tena Ruiz, Yvan Petillot, David M Lane, and Cedric Salson. Feature extraction and data association for auv concurrent mapping and localisation. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, volume 3, pages 2785–2790. IEEE, 2001.

[92] Andrew R Runnalls, Paul D Groves, and Robin J Handley. Terrain-referenced navigation using the igmap data fusion algorithm. In *Proceedings of the 61st Annual Meeting of the Institute of Navigation*, pages 976–986, 2005.

[93] M Jordan Stanway. Water profile navigation with an acoustic doppler current profiler. In *OCEANS'10 IEEE SYDNEY*, pages 1–5. IEEE, 2010.

[94] Luke Stutters, Honghai Liu, Carl Tiltman, and David J Brown. Navigation technologies for autonomous underwater vehicles. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(4):581–589, 2008.

[95] Niko Sünderhauf and Peter Protzel. Switchable constraints for robust pose graph slam. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1879–1884. IEEE, 2012.

[96] Niko Sünderhauf and Peter Protzel. Switchable constraints vs. max-mixture models vs. rrr-a comparison of three approaches to robust pose graph slam. In *2013 IEEE International Conference on Robotics and Automation*, pages 5198–5203. IEEE, 2013.

[97] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.

[98] Sebastian Thrun and Michael Montemerlo. The graph slam algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006.

[99] Nikolas Trawny, Stergios I Roumeliotis, and Georgios B Giannakis. Cooperative multi-robot localization under communication constraints. In *2009 IEEE International Conference on Robotics and Automation*, pages 4394–4400. IEEE, 2009.

[100] Angelos Tzotsos, Konstantinos Karantzalos, and Demetre Argialas. Object-based image analysis through nonlinear scale-space filtering. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(1):2–16, 2011.

[101] Rudolph Van Der Merwe et al. *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models.* PhD thesis, OGI School of Science & Engineering at OHSU, 2004.

[102] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[103] Sarah E Webster, Ryan M Eustice, Hanumant Singh, and Louis L Whitcomb. Advances in single-beacon one-way-travel-time acoustic navigation for underwater vehicles. *The International Journal of Robotics Research*, 31(8):935–950, 2012.

[104] N Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications.* MIT Press, Cambridge, 1949.

[105] Stefan Williams, Gamini Dissanayake, and Hugh Durrant-Whyte. Towards terrain-aided navigation for underwater robotics. *Advanced Robotics*, 15(5):533–549, 2001.

[106] Oliver J Woodman. An introduction to inertial navigation. Technical report, University of Cambridge, Computer Laboratory, 2007.

[107] HAN Yurong, WANG Bo, DENG Zhihong, and FU Mengyin. Point mass filter based matching algorithm in gravity aided underwater navigation. *Journal of Systems Engineering and Electronics*, 29(1):152–159, 2018.

[108] Pingping Zhu, Jason Isaacs, Bo Fu, and Silvia Ferrari. Deep learning feature extraction for target recognition and classification in underwater sonar images. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2724–2731. IEEE, 2017.