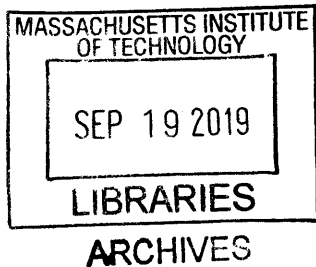# Dense, Sonar-based Reconstruction of Underwater Scenes

by

Pedro Nuno Vaz Teixeira

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and

WOODS HOLE OCEANOGRAPHIC INSTITUTION

September 2019

© Pedro Nuno Vaz Teixeira, MMXIX. All rights reserved.

The author hereby grants to MIT and WHOI permission to reproduce and to
distribute publicly paper and electronic copies of this thesis document in whole
or in part in any medium now known or hereafter created.

Author . . . .
Signature redacted

Department of Mechanical Engineering
August 20, 2019

Certified by . . . .
Signature redacted

John J. Leonard
Samuel C. Collins Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Accepted by . . . .
Signature redacted

David Ralston
Associate Scientist with Tenure, Applied Ocean Physics and Engineering
Chair, Joint Committee for Applied Ocean Science & Engineering Department
Committee on Graduate Theses

Accepted by . . . .
Signature redacted

Nicholas Hadjiconstantinou
Professor of Mechanical Engineering
Chair, Mechanical Engineering Department Committee on Graduate Theses

Dense, Sonar-based Reconstruction of Underwater Scenes

by

Pedro Nuno Vaz Teixeira

Submitted to the Department of Mechanical Engineering
on August 20, 2019, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Three-dimensional maps of underwater scenes are critical to—or the desired end product of—many applications, spanning a spectrum of spatial scales. Examples range from inspection of subsea infrastructure to hydrographic surveys of coastlines. Depending on the end use, maps will have different accuracy requirements. The accuracy of a mapping platform depends mainly on the individual accuracies of *(i)* its pose estimate in some global frame, *(ii)* the estimates of offsets between mapping sensors and platform, and *(iii)* the accuracy of the mapping sensor measurements. Typically, surface-based surveying platforms will employ highly accurate positioning sensors—*e.g.* a combination of differential global navigation satellite system (GNSS) receiver with an accurate attitude and heading reference system—to instrument the pose of a mapping sensor such as a multibeam sonar.

For underwater platforms, the rapid attenuation of electromagnetic signals in water precludes the use of GNSS receivers at any meaningful depth. Acoustic positioning systems, the underwater analogues to GNSS, are limited to small survey areas and free of obstacles that may result in undesirable acoustic effects such as multi-path propagation and reverberation. Save for a few exceptions, the accuracy and update rate of these systems is significantly lower than that of differential GNSS. This performance reduction shifts the accuracy burden to inertial navigation systems (INS), often aided by Doppler velocity logs. Still, the pose estimates of an aided INS will incur in unbounded drift growth over time, often necessitating the use of techniques such as simultaneous localization and mapping (SLAM) to leverage local features to bound the uncertainty in the position estimate.

The contributions presented in this dissertation aim at improving the accuracy of maps of underwater scenes produced from multibeam sonar data. First, we propose robust methods to process and segment sonar data to obtain accurate range measurements in the presence of noise, sensor artifacts, and outliers. Second, we propose a volumetric, submap-based SLAM technique that can successfully leverage map information to correct for drift in the mapping platform's pose estimate. Third, and informed by the previous two contributions, we propose a dense approach to the sonar-based reconstruction problem, in which the pose estimation, sonar segmentation and model optimization problems are tackled simultaneously under the unified framework of factor graphs. This stands in contrast with the traditional approach where the sensor processing and segmentation, pose estimation, and model reconstruction problems are solved independently. Finally, we provide experimental results obtained over several deployments of a commercial inspection platform that validate the proposed techniques.

Thesis Supervisor: John J. Leonard
Title: Samuel C. Collins Professor of Mechanical and Ocean Engineering

Acknowledgements

I would first like to thank Prof. Franz Hover for accepting me into the Joint Program, and for his mentorship and advice, which was certainly fundamental in successfully navigating the first years of graduate school. Over classes and lunches, research and trips, experiments and long conversations, Brooks, Chris, Eric, Josh, Lucille, and Mei made for great colleagues and friends.

I am also very grateful to Prof. John Leonard for taking me into the Marine Robotics group while simultaneously handling his sabbatical leave. At the Marine Robotics group I have learned tremendously and gained great insight from the discussions and conversations with people in the group. I am particularly grateful to Brendan, David, Kevin, Kurran, Mark, Ross, Roxana, Sudeep, and Tonio.

Between "showing me the ropes" on the first HAUV experiments and the many research discussions over the phone, Prof. Michael Kaess has been a steadfast source of advice and support over the years, for which I am truly thankful. Prof. Hanu Singh's knowledge and expertise have been very valuable, and the occasional field story has made many classes and meetings much more captivating and entertaining. I have also been fortunate for the opportunity to spend a Summer at Prof. Erin Fischell's lab at WHOI, and to witness her talent and enthusiasm. She has been a very welcome addition to my thesis committee.

The HAUV field experiments would not have been possible without the support of Tamara Kick and Greg Garvey at ONR, as well as the Navy divers and boat pilots who ensured things went smoothly, even when they were not going as planned. My time with the HAUV project was a lot more fun and enriching—both in and out of the boat—thanks to the people I had the pleasure of working with: Prof. Ryan Eustice, James, Jie, Josh, Paul, and Steve at PeRL and Bing, Eric, and Paloma at RPL.

There are many others both at MIT and beyond who I must also acknowledge for their friendship and support: Oscar, Sam, Stephen, Tim, Thomas and Narges certainly come to mind. With Misha and Paul around, the Marine Autonomy lab has been a great place to work and hangout. I have really enjoyed the advice and conversations with Derek and Ron at CSAIL. Kris, Lea, Leslie, and Nira kindly guided me through the formal side of the program. Jamel, Yan, Sergio and Liz have always made me feel at home. The time spent with Pedro over brief and sporadic visits back home has felt as if I had always been there. Dehann and Nick have been great friends and colleagues from day one, and have made my time at MIT even more enriching, both personally and professionally. Between "talking shop", discussing research over yet another coffee, or going for a quick bite, it really has been a pleasure spending time with them.

Over the years, my parents—Cristina and Joaquim—provided me with invaluable support and encouragement,for which I am forever grateful. Rui and my other "little brother" Sam have teased me as much as they have cheered me on, and brought me great joy in

the process. The weekly videoconferencing, brief chats, and unexpected phone calls made the distance seem shorter and the problems smaller. This journey would also not have been possible without Joana. Despite my ramblings, lame attempts at humor, and going through many of the same motions of grad school at the same time as I, she always has time to listen, advise, encourage, and support—all while being an amazing human being. Thanks to her, I am also lucky to have Ana, Pedro, and Margarida in my family.

Contents

List of Figures

List of Tables

14

List of Acronyms

AHRS Attitude and Heading Reference System

AUC area under curve

AUV Autonomous Underwater Vehicle

AWGN additive white Gaussian noise

DIDSON Dual-frequency Identification Sonar

DVL Doppler velocity log

EKF extended Kalman filter

GNSS Global Navigation Satellite System

HAUV Hovering Autonomous Underwater Vehicle

ICM Iterative Conditional Modes

ICP Iterative Closest Point

IHO International Hydrographic Organization

IMU Inertial Measurement Unit

INS Inertial Navigation System

KLD Kullback-Leibler divergence

LBL Long Baseline

MAP maximum a-posteriori

MBES Multibeam Echo Sounder

MRF Markov Random Field

NTP   Network Time Protocol

PPS   Pulse-per-second

PTP   Precision Time Protocol

ROC   Receiver Operating Characteristic

ROV   Remotely Operated Vehicle

SA   Simulated Annealing

SAS   Synthetic Aperture Sonar

SLAM   Simultaneous Localization and Mapping

TVG   time-varying gain

USBL   Ultra-Short Baseline

WHOI   Woods Hole Oceanographic Institution

Notation

Variables

x - scalar

$\mathbf{x}$ - vector (assumed column vector)

X, or $\mathbf{X}$ - matrix

$\mathcal{X}$ - set

Operators

$\tilde{\cdot}$ - measurement

$\hat{\cdot}$ - estimate

$\bar{\cdot}$ - homogeneous representation ($\bar{\mathbf{x}} = [\mathbf{x}^T\ 1]^T$)

Reference Frames

$^F\mathbf{p}$ - a *point* whose coordinates are expressed in the reference frame F

$^F\mathbf{x}$ - a *pose* whose coordinates are expressed in the reference frame F

$^W_F T$ - homogeneous transformation matrix, describing the transformation from F to W ($^W\bar{\mathbf{p}} = {}^W_F T\, {}^F\bar{\mathbf{p}}$).

Whenever an orientation of a pose is expressed in terms of its *yaw*, *pitch*, and *roll* (ψ, θ, ϕ), these follow the rotation sequence Z-Y-X.

Chapter 1

Introduction

1.1 Motivation

Hidden under the commonly referenced adage that *"we know more about the surface of the Moon than we do about the ocean floor"* lies what is still a great unknown. From shipwrecks to mountains, many important features remain unresolved in most maps of the ocean floor. As of this writing, full bathymetric coverage of the ocean basins has only been achieved at a spatial resolution of 5 kilometers, using a combination of highly accurate satellite altimetry and gravimetric models. The General Bathymetric Chart of the Ocean's *Seabed 2030* project, which plans to create high resolution maps of the world's oceans by the year 2030, aims for 93% coverage for depths greater than 200 meters at a spatial resolution of 100 meters. Using current methods, this task is expected to require 350 years of ship time at a cost in the order of billions of dollars [32]. This stands in stark contrast with most land maps: the TerraSAR/TanDEM-X satellite formation, for instance, uses synthetic aperture radar to achieve a vertical accuracy of 4 meters at a resolution of 12 meters, with revisit periods of less than two weeks. Civilian optical satellites achieve even more impressive results, with the Pléiades constellation attaining 0.5 meters spatial resolution. If ocean basins are to be mapped at a comparable resolution (10 meters), then the need for high-resolution, autonomous mapping technology is clear: with the ability to operate at depths of up to 6000 meters[1] for over a day, Autonomous Underwater Vehicles (AUVs) can cover large swaths of terrain without the presence of a support ship other than for launch and recovery. Survey operations performed by an AUV fleet, and supported by a single ship, have been demonstrated successfully in the past few years, pointing in a very promising direction to obtain such high resolution maps.

---

[1]This is currently the highest depth rating on a commercial AUV system (Kongsberg's Hugin and Remus 6000 AUVs), allowing access to more than 95% of the world's ocean basins.

Before proceeding further, however, we must first support our emphasis on the need for high resolution; after all, uses for very accurate, high resolution maps of abyssal plains, for instance, may not be immediately obvious. The *International Hydrographic Organization (IHO) Standards for Hydrographic Surveys*, for instance, require a maximum horizontal uncertainty of 2 meters at a 95% confidence level for its most strict survey grade, but these are only required in certain shallow water areas [94]. Since the end-use of most of these surveys is the elaboration of nautical charts, this is deemed sufficient accuracy for navigation purposes. Still, there are many important applications where the accuracy and resolution needs are much stricter, of which we list a few examples below.

1.1.1   Environmental Monitoring

Nuclear Waste Dump Sites

Between 1946 and 1993 the amount of radioactivity from radioactive waste dumped in the oceans reached a maximum of 4.5×10^4 TBq[2]; it has since decreased (through decay) to less than half of this value. The waste dumped at these sites ranges from low-level solid waste to full nuclear reactor vessels with spent fuel; the northeast Atlantic alone contains well over 100,000 tons of waste containers [40]. In addition, there are also radioisotope thermoelectric generators, nuclear warheads, and reactors (in the wrecks of nuclear-powered military vessels) whose locations are not known precisely [39]. Locating and monitoring these sites requires the use of platforms that can withstand full ocean depth (6000 meters) and produce maps with sufficient resolution to resolve small scale features, such as corrosion on a nuclear warhead or waste container.

1.1.2   Explosive Ordinance Disposal (EOD)

Detection and removal of explosive devices such as limpet mines is an important task for navies worldwide. This is often accomplished through a combination of trained marine mammals, divers, and small remotely operated vehicles, all of which have considerable drawbacks. The deployment of divers and/or trained marine mammals exposes them to very hazardous environments and, with the latter, it is often hard to ensure complete hull coverage. Similarly, while small ROVs are agile, they have very limited navigation and manipulation capabilities, which hinders their ability to localize and neutralize a device.

Unexploded ordinance (UxO) disposal is not limited to active ships, as exemplified by the *SS Richard Montgomery*. This *Liberty*-class cargo ship was wrecked in the Thames Estuary during World War II while carrying munitions. The 1400 tons of explosive cargo

---

[2]1 TBq = 1×10^{12}Bq - *becquerel*, a unit of radioactivity.

that remain in the wreck and its proximity to inhabited areas pose a significant hazard, which has led to the creation of an exclusion zone around the wreck, as well as periodic surveying to monitor its condition [97].

1.1.3 Inspection, Maintenance, and Repair (IMR)

One of the underwater mapping applications for which there is growing commercial demand is the inspection, maintenance, and repair of subsea infrastructure. These applications comprise a variety of tasks, such as inspecting long pipelines that connect wells to manifolds, surveying "christmas trees", and operating equipment [5, 57]. At the present, most of these tasks are performed using Remotely Operated Vehicles (ROVs), which require skilled pilots and the deployment of expensive support ships. This has sparked the development of inspection platforms that can perform some of these tasks autonomously, thereby reducing some of the major costs associated with IMR operations [16].

1.1.4 Search and Rescue (SAR)

The last ten years have witnessed a few high profile accidents where AUVs, ROVs, and other underwater mapping platforms have played a crucial role in the search and recovery efforts.

Air France flight 447

Air France flight 447 crashed into the Atlantic Ocean on June 1st, 2009, after the airplane entered a high-altitude stall condition from which it did not recover. Efforts to find the missing airplane and its black boxes began almost immediately, but proved unfruitful, with the third search phase ending in late May of the following year [102]. The debris field was finally identified during a fourth search phase in early April 2011, from side-scan sonar data obtained by REMUS 6000 AUVs operated by the Woods Hole Oceanographic Institution (WHOI). A second pass over the debris field provided photographic coverage which was critical to the successful recovery of both the flight data and cockpit voice recorders by an ROV [28].

Malaysia Airlines flight 370

The circumstances behind the disappearance of Malaysia Airlines Flight 370 on March 8, 2014, remain unknown[3], despite the most costly search effort in history, involving a variety of assets and equipment from nine different countries. This search was also notable

---

[3]The description of the search for MH370 is based on the known facts as of the time of this writing.

for highlighting the lack of detailed maps in many areas of the ocean. Specifically, areas for which the only available maps had a resolution of 5 kilometers per pixel, have now been mapped using high resolution multibeam echo sounders and synthetic aperture sonars, which can resolve features that are 100 times smaller [79]. In the process, four shipwrecks were found, as were topological features as large as underwater volcanoes [107, 125]. Given the configuration of the search area and the harsh sea conditions faced by the survey vessels, sonar-equipped towed bodies were the platform of choice for mapping operations. AUVs were a key component of the last search phase, in which the survey company *Ocean Infinity* deployed a fleet of eight *Kongsberg Hugin* AUV to cover in excess of 100,000 square kilometers. The same AUV fleet has also been deployed to find the wrecks of South Korean tanker *Stellar Daisy* and Argentinian submarine *ARA San Juan*.

Having motivated the need for high resolution mapping, we must now move on to the key factors behind it. The accuracy of a map depends mainly on the individual accuracies of:

(i) the estimate of the position and orientation (pose) of the mapping in some global frame,

(ii) the estimates of the offsets between the mapping sensors, the platform, and its navigation sensors, and

(iii) the mapping sensor measurements.

In the following sections, we provide some examples of the typical navigation and mapping payloads used in underwater reconstruction applications.

1.2 Navigation Sensors

For underwater platforms, the rapid attenuation of electromagnetic (EM) signals in water precludes the use of Global Navigation Satellite System (GNSS) receivers at any meaningful depth. Underwater analogues to GNSS—acoustic positioning systems such as Long Baseline (LBL) and Ultra-Short Baseline (USBL) —can attain meter-level (or better) accuracy in platform position estimates, but have significant drawbacks [77]. LBL requires the deployment of a minimum number of beacons, which need to be surveyed before they can be used for positioning. USBL, on the other hand, requires a surface-deployed transponder with a known position—often a GNSS-equipped support-ship. The attenuation of sound in water and its comparatively slow speed (with respect to EM waves) places limits on both the coverage and update rate offered by these systems. Finally, it is also important

to note that when used in acoustically complex environments, these systems are subject to undesirable phenomena such as occultation, reverberation and multi-path propagation, which can significantly degrade their positioning performance.

Nearly all inspection and mapping platforms rely on some form of *dead-reckoning* as their primary source of pose estimates. This often takes the form of a Doppler velocity log (DVL)-aided Inertial Navigation System (INS), in which the body-relative bottom velocity is first transformed to and integrated in the local-level frame (described in section 1.5) using an attitude estimate from an on-board Attitude and Heading Reference System (AHRS)[4] [11, 47, 48, 49, 27, 121]. The performance of these systems, often described by the horizontal position uncertainty as a percentage of distance traveled, is mostly determined by the accuracy of the velocity and yaw measurements. Noise in the velocity estimate will cause an unbounded growth (random walk) in the position estimate; this is made worse by the drift in the yaw estimate found in most INSs. One of the main limitations of dead-reckoning DVL-aided INSs is that bottom-relative velocity measurements are only available below a certain altitude (distance from the seafloor); above this, the DVL can provide velocity measurements relative to the water mass, which requires estimating water column velocity to obtain a global position estimate. This is often accomplished through the use of kinematic and dynamic vehicle models [33].

1.3   Mapping Sensors

Most of the sensors used in underwater mapping platforms fall in one of two categories: *acoustical* and *optical*. While the former have long been the backbone of most, if not all, mapping efforts, recent advances in underwater optical systems have enabled the deployment of inspection platforms with metrology-grade lidar systems.

1.3.1   Optical Mapping Sensors

Optical mapping sensors include camera-based, structured light, and lidar systems. These sensors offer very high resolution (1-10 millimeters), but often at the expense of power or range: between the lighting systems required by cameras and the high-intensity lasers used in both structured light and lidar sensors, it is not uncommon for optical systems to require power in the range of 50 to 150 watt. At the same time, the range of these sensors is limited by the water conditions: while clear water environments can allow for ranges

---

[4]Like an INS, an AHRS has an inertial measurement unit at its core; the difference between the two being that an AHRS is used exclusively for orientation estimates, whereas an INS is used for both position and orientation estimates. More advanced INS may also estimate body velocity, local gravity vector and other parameters.

in excess of 40 meters for high-power lidars, turbid environments such as harbors and coastal areas can see the maximum range of such sensors decrease to a few meters. As both structured light and lidar sensor are prohibitively expensive for all but high-end inspection mapping platform, camera-based systems are the most common of the three, with stereo cameras proving a very cost-effective option for high-resolution optical mapping.

1.3.2 Acoustic Mapping Sensors

Acoustic mapping sensors have evolved from simple pencil-beam echo sounders to a wide variety of sensors, including three-dimensional imaging systems and Synthetic Aperture Sonars (SASs). The most commonly used sonars for mapping, however, are side-scan sonars and Multibeam Echo Sounders (MBESs). The former can often be found in towed bodies and torpedo-shaped AUVs, as their wide swath combined with the platform velocity produces high coverage rates. Because of their inability to capture terrain geometry, they are often complemented by profiling[5] multibeam sonars. As with all sonars, the attenuation of sound forces multibeams to operate on a trade-off between resolution and range: low-frequency sonars can reach distances beyond 10 kilometers at low resolution; high frequency models attain centimeter resolution but are often limited to distances of up to a few tens of meters. One of the main advantages of high-frequency multibeam sonars, despite their limited range, is their robustness to turbidity, which allows them to operate in high turbidity environments, such as the ones mentioned above. Advances in sensor technology, and the combination of Moore's law and Dennard scaling[6] have enabled the appearance of the first real-time three-dimensional sonars [17, 64].

As most of the research in this dissertation was motivated by the problem of ship hull inspection, it is safe to assume that the water conditions encountered by an inspection platform will often be turbid—this has certainly been the case in all the data collection deployments for which results are presented. The short visual range (usually in the range of 1-2 meters, but falling below 1 meter in certain conditions), combined with the need to operate safely when inspecting geometrically complex areas, such as the running gear of a large ship, motivates the use of sonar for a major part of the mapping and inspection

---

[5]A *profiling* multibeam has a narrow vertical field of view, and is pointed perpendicularly to the terrain, producing a scan that is similar to a "slice" of the scene along the scanning plane. An *imaging* multibeam has a wider vertical field of view (tens of degrees), and is pointed at an angle, yielding scans that are similar to those of a camera.

[6]While Moore's law describes the doubling in the number of transistors (in an integrated circuit) every two years brought about by miniaturization, Dennard scaling states that their power density remains constant - the combination of the two implies that performance per watt doubles every two years. This allows for the computational resources in both sensors and platforms to grow at the same rate while keeping the same power budget.

tasks. For this reason, we will focus on the high-frequency profiling multibeam sonar as the primary mapping sensor. This does not preclude the use of other sensors; in fact, the techniques described in the following chapters can easily be extended to work using data from structured light or lidar sensors.

1.4 Related Work

Most of the work in high-resolution underwater mapping can be split into one of two categories: *featureless* and *feature-based* approaches, depending on their use of environmental features to describe the scene. The ubiquity of multibeam echo sounders, in combination with the ambiguities in both camera and side-scan features (as well as the limited computational resources available to identify, track, and match such features) meant that early work on underwater mapping was often accomplished using featureless approaches. As both sensor and computer technology develop, however, feature-based techniques are becoming ever more popular and successful.

1.4.1 Feature-based techniques

Feature-based techniques are commonly found in methods using *imaging*-type modalities, such as cameras, imaging, and side-scan sonars [23, 24, 41, 80, 81]. The main challenges faced by these approaches are the feature sensitivity to the ensonification (sonar) or lighting (camera) conditions, as well as the ambiguity associated with the location of the feature with respect to the sensor, as both cameras and imaging sonar provide under-constrained measurements.

1.4.2 Featureless techniques

When the primary mapping sensor provides data that is closer to a range measurement, the use of featureless techniques is more prevalent. This is the case with profiling multibeam echo sounders, as well as other sonars where the narrow beam widths allow for small azimuth/elevation ambiguities in range measurements.

Much of the early research on simultaneous localization and mapping in underwater environments has addressed the problem from a two-dimensional perspective, where the platform pose comprises its position and orientation in the horizontal plane. Many of these techniques employ single-beam, mechanically-scanned imaging sonars, whose 360° coverage makes them well-suited to some of the scan-matching techniques that have been proven successful in land applications [54]. Unlike the lidars used by their terrestrial counterparts, the scanning sonars used by these underwater robots have cycle times that are

comparable to, and often slower than, the vehicle dynamics. To avoid the motion-induced artifacts that result from this limitation, some of the techniques will either assume—or keep—the vehicle in-place while a full scan is assembled [61]. Platforms equipped with dead-reckoning sensors relax this operational limitation by relying on the pose estimate that these sensors provide to compensate for vehicle motion while the scan is assembled [58, 59].

Extending featureless techniques to three dimensions shares some of the challenges with the previously described planar mapping techniques: just as these have to address the constraints imposed by the scanning sonar, the same holds for many three dimensional mapping methods. This limitation, as described earlier in this chapter, stems from the operational constraints imposed by profiling multibeam sonars, as attaining reasonable coverage rates requires moving perpendicularly to the scanning plane, which removes any overlap between scans, hence precluding the direct use of scan-matching techniques. To circumvent this limitation, the standard approach is to assemble maps over small spatial scales using a limited number of scans. These *submaps*, similar to virtual 3D sonar measurements, can then be pairwise registered. Such techniques have been successfully demonstrated in microbathymetric applications [83, 84, 82, 85], where the 2.5D nature of the scene (elevation model) allows the submaps to be treated as images and leverage image registration methods.

Some sensor and platform configurations can avoid the need to create submaps: the volumetric range measurements produced by 3D sonars allow for direct pairwise registration of scans [65]. The DEPTHX robot, for instance, was equipped with 54 pencil-beam sonars arranged in three perpendicular rings, allowing for partial position measurements with respect to a current map estimate to be obtained. Using occupancy grids as its map representation, this platform was used to produce maps of flooded sinkholes with a resolution of a few tens of centimeters [26]—this representation, well-suited for use with sonar, had also been used two decades earlier in some of the first ship- and ROV-based mapping efforts [100]. DEPTHX's highly-accurate gyro-compass and depth sensor pair allowed the orientation and depth measurements to be considered as drift-free estimates, thereby simplifying the full pose estimation problem to that of estimating the platform's horizontal position, which was accomplished using a particle filter.

1.4.3   Hybrid approaches

It is also worth noting that there exist hybrid approaches, combining both feature-rich camera data with the ranging measurements obtained from multibeam sonar [50, 51, 70, 71, 73, 74, 72, 75]. In one of the earliest approaches combining the two modalities, a sonar is used to obtain the terrain profile and initialize the extraction of visual features [56, 122].

These features are added to an extended Kalman filter (EKF) Simultaneous Localization and Mapping (SLAM) framework, and tracked between successive frames. The use of sonar information reduces the ambiguity in the visual features, and allows for the estimation of elevation and azimuth measurements for each tracked feature. These methods are demonstrated experimentally by the mapping of sections of coral reef.

1.5 Reference Frames

The main reference frames used in this work are [27, 68]:

- **World frame** (W) - this is the global, Earth-fixed frame, which follows the NED (North-East-Down) convention for the orientation of its axes.

- **Local-level frame** (L) - also known as the tangent, or vehicle-carried frame, this frame has its axes parallel to the world frame, but its origin is coincident with that of the platform frame.

- **Platform frame** (P) - the platform or body-fixed frame is located at some reference point in the platform, and its axes follow the Forward-Port-Down convention.

- **Navigation frame** (N) - this is the reference frame used by the navigation package, whose estimate describes the transformation from the world to this frame. Throughout the text we assume that the offset between the navigation and platform frames are known and accounted for, so that the pose estimates describe the pose of the *platform* frame in the world frame.

- **Sensor frame** (S) - similarly to the navigation frame, this is the reference frame used by the mapping sensor. Depending on the platform configuration, several of these frames may exist (e.g. camera, multibeam, lidar).

1.6 Assumptions

The main assumptions made in the course of this work are:

- *A pose estimate is available from the platform* - as mentioned in section 1.2 most, if not all mapping platforms, carry some form of aided inertial navigation system whose short-term accuracy is is sufficiently high to obtain small scale maps; the consequence of this assumption is that we can build upon these estimates to focus on the reconstruction problem and on how to address the medium- and long-term drift that these

systems exhibit. Moreover, by not requiring a specific navigation architecture, the solutions proposed in chapters 4 and 5 can also be extended to incorporate individual navigation sensor measurements. Finally, it is worth noting that this assumption is commonly made in the underwater mapping literature [24].

- *The travel time of the sonar signal can be considered instantaneous* - In order to produce high resolution maps, mapping platforms will employ high-frequency (>1 megahertz) multibeam echo sounders. Due to the high attenuation at these frequencies, the effective range of these sonars is often under 10 meters, which corresponds to a two-way travel time of less than 3 milliseconds. Given the typical mapping platform moves at speeds of less than 2 meters per second, the effects of the finite travel time are assumed negligible.

- *Sound refraction can be neglected* - the relatively small volume ensonified by the multibeam echo sounder is considered well-mixed and sufficiently uniform so that the effects of refraction due to water density variations are negligible.

- *All sensors are synchronized* - a modern underwater mapping platform will often contain a network of computers: most sensors will carry their own embedded system and communicate with the main computer, where data is stored, processed and/or relayed, via some combination of serial or network protocol. While most of these systems will not be equipped with accurate clocks, many solutions exist to keep them in relative synchrony. Both the Network Time Protocol (NTP) and the Precision Time Protocol (PTP), as well as on-board Pulse-per-second (PPS) signal distribution and triggering schemes can be used to keep timing differences between systems to within milli- or even micro-seconds.

- *Sensor offsets have been calibrated prior to platform deployment* - chapter 2 describes the three dominant factors in reconstruction accuracy; while sensor offsets are one of them, they are often estimated to a sufficient degree of accuracy during pre-mission operations and considered fixed for the remainder of the mission.

- *The accuracy limitations of a local Cartesian coordinate frame are negligible* - we assume that the scene is small enough that the limitations of a local Cartesian frame can be considered negligible with respect to other sources of uncertainty.

Chapters 3 through 5 will describe the motivations for these assumptions and, where necessary, introduce other minor assumptions. Section 6.2.1 will revisit the main assumptions and provide remarks on how some of these can be relaxed.

1.7 Contributions

The contributions presented in this dissertation aim at improving the accuracy of maps of underwater scenes produced from multibeam sonar data. The main contributions are summarized below:

- First, we propose robust methods to process and segment sonar data to obtain accurate range measurements in the presence of noise, sensor artifacts, and outliers [105]. These methods attempt to model acoustic phenomena such as attenuation and beam pattern to recover an improved estimate of the sonar data that is then fed to either a *dense* or *sparse* segmentation technique. In the dense approach, a label is computed for every range-azimuth cell in the scan from an intensity distribution model estimated online; in the sparse approach, a label is estimated for every beam.

- Second, we propose a volumetric, submap-based SLAM technique that can successfully leverage map information to correct for drift in the mapping platform's pose estimate [104]. *Submaps* are small, self-consistent maps obtained by grouping together scans over short periods of time, which enable the use of scan-matching techniques to derive loop closures. The reduction in the number of poses brought about by the use of submaps also allows the proposed technique to be used in real-time applications.

- Third, and informed by the previous two contributions, we propose a dense approach to the sonar-based reconstruction problem, in which the pose estimation, sonar segmentation and model optimization problems are tackled simultaneously under the unified framework of factor graphs [106]. This stands in contrast with the traditional approach where the sensor processing and segmentation, pose estimation, and model reconstruction problems are solved independently.

For each of the techniques described above, we provide experimental results that validate the proposed methods. These were obtained over several deployments of a commercial inspection platform, under the scope of the Office of Naval Research's ship hull inspection project.

1.8 Overview

This dissertation is structured as follows: Chapter 2 motivates and describes the overarching mapping and pose estimation problems in a more formal fashion. Chapter 3 addresses the processing and segmentation of data from the primary mapping sensor—the multibeam echo sounder—to obtain measurements that can be used to create a model; chapter

4 builds upon these results and describes a submap-based technique that can be used to produce maps in real time while simultaneously mitigating the effects of the drift in the position estimates from the on-board navigation system. Chapter 5 aims at bringing the two problems together under a unified framework to not only address some of the key limitations of the technique presented in chapter 4, but also to leverage the problem of model optimization to improve the accuracy in both the sensor range and pose estimates. Finally, chapter 6 offers some concluding remarks on the proposed techniques, revisits some of the key assumptions made in the process, and describes promising directions for future work.

Chapter 2

Problem Statement

2.1 Introduction

This chapter describes the problem of high resolution mapping, to arrive at a model for the mapping error/uncertainty as a function of its three main components: pose estimate uncertainty, sensor offset uncertainty, and sensor measurement uncertainty.

2.2 Problem Formulation

The problem of producing an accurate map of a scene can be stated as one where we wish to spatially register sensor data in the world frame. Depending on the intended use of the map, it may have to fulfill certain requirements - navigation charts, for instance, must comply to the IHO's Special Publication 44 [94]. This standard specifies minimum accuracy requirements for different survey classes, which are expressed in terms of the maximum allowable values (2σ) for the horizontal and vertical uncertainties in the positioning of soundings, measurements, aids to navigation, and other relevant features.

Assuming the primary mapping sensor measures the position of some feature or object point in its own frame ${}^{s}x$, that position can be registered in the world frame by

$$^{w}x = {}^{w}_{p}T\,{}^{p}_{s}T\,{}^{s}\bar{x} \tag{2.1}$$

where ${}^{w}_{p}T$ is the platform's pose in the world reference frame, and ${}^{p}_{s}T$ is the sensor pose in the platform frame. As previously mentioned, map accuracy is a function of the uncertainty associated with the position of this feature or point in the world frame which, in turn, depends on the uncertainty associated with each of the terms on the right hand side of equation (2.1).

In practical applications there are often more reference frames at play, with at least one for each sensor used in navigation (e.g. DVL, Inertial Measurement Unit (IMU), pressure sensor, ...) or mapping (e.g. cameras, lidar, multibeam sonar, ...). As some of these will be mounted on actuators, such as a pan/tilt mount, the pose of some of these frames with respect to the platform may vary over time. Since these frames are tied to the actual configuration of the mapping platform, we will instead consider a more general model using just the following three reference frames: *world*, *platform*, and *sensor*. In the next section we will propose simple models for how the uncertainty in the estimates associated with these frames impacts mapping performance over time.

2.2.1 Sensor Performance

In equation (2.1), we assumed the mapping sensor provided the position of an object point or feature in its own frame, $^s x$. In fact, it is often the case that these sensors instead provide a range measurement to that point or feature, $\tilde{r}$, along a nominal measurement direction $\hat{b}$. While this "single measurement" sensor model may appear overly simplistic, it can easily be extended to describe more typical sensors outputting a set of such measurements, $(\tilde{r}_i, \hat{b}_i)$, such as sonars or lidars.

Assuming an unbiased sensor, we can model the range measurement as

$$\tilde{r}_p = r_p + \nu_r \tag{2.2}$$

where the measurement noise ν_r will follow some probability distribution which, for the sake of this discussion, we will approximate as a zero-mean normal distribution with standard deviation σ_r[1]. This range measurement is projected along the sensor's nominal pointing direction $\hat{b}$ to register the measurement in the sensor's reference frame:

$$^s \tilde{x}_p = \hat{b} \tilde{r}_p. \tag{2.3}$$

Depending on the nature of the sensor, there will be a varying degree of uncertainty on the direction $\hat{b}$, which may be captured by the degree of accuracy in the sensor's calibration, or by some other sensor property such as the beam pattern for acoustic or electromagnetic transducers such as sonar or radar. Parameterizing the unit vector's orientation with *azimuth* and *elevation* angles α and β, again subject to some normally-distributed noise

---

[1]As expected, range measurement noise is likely to follow some other distribution that is significantly different from the normal distribution, and likely multimodal. One such example can be found in acoustics-based range measurements, where phenomena such as refraction and multi-path can significantly skew range measurements while also introducing multiple modes in the range distribution.

around their nominal values

$$\tilde{\alpha} = \alpha + v_\alpha$$
$$\tilde{\beta} = \beta + v_\beta$$

(2.4)

we can express the nominal pointing direction as

$$\hat{b} = \begin{bmatrix} \cos\tilde{\beta}\cos\tilde{\alpha} \\ \cos\tilde{\beta}\sin\tilde{\alpha} \\ \sin\tilde{\beta} \end{bmatrix}.$$

(2.5)

Under these assumptions, the position of the range measurement in the sensor frame will follow a normal distribution centered around the true position, $^S\mu_P = {}^s x = \hat{b}r$, and with covariance $^S\Sigma_p = J\Sigma J^T$, where J is the Jacobian of the projection described by equation (2.3) [96]

$$J = \begin{bmatrix} \cos\alpha\cos\beta & -r\sin\alpha\cos\beta & -r\cos\alpha\sin\beta \\ \sin\alpha\cos\beta & r\cos\alpha\cos\beta & -r\sin\alpha\sin\beta \\ \sin\beta & 0 & r\cos\beta \end{bmatrix}$$

(2.6)

Note that this is an approximation for the true distribution associated with $^s x_p$, and is only valid for small values of σ_α, σ_β; if large deviations from the nominal measurement direction are expected, the resulting distribution cannot be reasonably approximated by a multivariate normal in Cartesian space.

2.2.2 Sensor Offsets

The position of the range measurement in the platform frame is

$$^P x_p = {}^P x_S + {}^P_S R {}^S x_p$$

(2.7)

where $^P x_S$ and $^P_S R$ are the sensor position and orientation in the platform frame, with associated uncertainties $^P\Sigma_P = diag([\sigma_{sx}^2 \ \sigma_{sy}^2 \ \sigma_{sz}^2])$ and $^P\Sigma_A = diag([\sigma_{s\psi}^2 \ \sigma_{s\theta}^2 \ \sigma_{s\phi}^2])$, respectively, using a *yaw* (ψ), *pitch* (θ), *roll* (ϕ) - YPR - attitude parameterization. Under the small angle assumption, the uncertainty in the position of the range measurement in the platform frame can be approximated by a normal distribution with mean $^P x_p$ and covariance

$$^P\Sigma_p = {}^P\Sigma_P + J_A \left({}^P\Sigma_A\right) J_A^T + {}^P_S R \left({}^P\Sigma_p\right) {}^P_S R^T$$

(2.8)

where J_A is the submatrix of the Jacobian for the transformation described by equation (2.7) comprising the partial derivatives with respect to orientation (again, under a YPR parameterization). For the special case where the estimate of the offset between platform

and sensor frames is sufficiently accurate that small angle perturbations can be considered $(\psi, \theta, \phi \approx 0 \implies {}^P_S R \approx I_{3 \times 3})$, J_B can be approximated by [8, p. 20-21]

$$J_A = \begin{bmatrix} -{}^S y_p & {}^S z_p & 0 \\ {}^S x_p & 0 & -{}^S z_p \\ 0 & -{}^S x_p & {}^S y_p \end{bmatrix} \tag{2.9}$$

The three terms on the right side of equation (2.8) correspond to the contributions due to uncertainty in sensor translation, orientation, and range measurement (respectively). The alignment assumption above makes it easier to see that, for large range measurements, uncertainty in sensor orientation, rather than position, is the dominant factor in registration accuracy.

2.2.3 Navigation Performance

To model the remaining component of the right-hand side of equation (2.1), we introduce a simple dead-reckoning model, based on a rate-integrating gyroscope and body-relative velocity measurements. This is a planar analogue to the DVL-aided inertial navigation systems found in many underwater platforms [48].

In this model, vehicle kinematics are limited to the horizontal plane and approximated by a simple integrator, where the state vector $x = [x \ y \ \psi]^T$ represents position (x, y) and heading ψ. The control input $u = [u \ v \ \omega_z]^T$ corresponds to the linear velocity in the horizontal plane (u, v), and angular velocity component along the platform's z axis, ω_z. The vector v, known as the process noise, accounts for uncertainty in the model, and is assumed to be zero-mean and normally distributed, with covariance Σ.

$$\dot{x} = u + v \tag{2.10}$$

Under this simple model, the uncertainty associated with the state estimate, P, will grow over time, as described by

$$\dot{P} = \Sigma \tag{2.11}$$

To produce a pose estimate, we drive the model described by equation (2.10) with an estimate of u. While this estimate can be derived from commanded values (set points), most estimators will instead rely on measurements $\tilde{u}$.

$$\dot{\hat{x}} = \tilde{u} \tag{2.12}$$

Over time, noise, bias, and other phenomena affecting these measurements will cause the

estimate $\hat{x}$ to diverge from the true state x - this is captured by the increase in the uncertainty associated with the state estimate, P. This growth is governed by models of the measurements used to drive the model, as explained below.

Angular speed measurement model

In the absence of an absolute heading reference, a heading estimate $\hat{\psi}$ can be computed by integrating angular velocity measurements coming from a gyroscope. These measurements are typically contaminated with additive white Gaussian noise (AWGN), v_{ω_z}, and a non-zero (uncompensated) bias, modeled as a random walk driven by process noise v_{b_z} (also AWGN).

$$\tilde{\omega}_z = \omega_z + b_z + v_{\omega_z}$$
$$\dot{b} = v_{b_z} \tag{2.13}$$

Linear velocity measurement model

Velocity sensors measure the platform velocity with respect to an external reference, such as the local terrain or the surrounding water column. These measurements are also assumed to be contaminated with AWGN v_u.

$${}^b\tilde{u}_b = {}^b u_b + v_u \tag{2.14}$$

Using the heading estimate $\hat{\psi}$, these velocity measurements are projected onto the global reference frame (${}^w u_b = {}^w_b R \, {}^b u_b$) and integrated to obtain the horizontal position estimate $[\hat{x} \; \hat{y}]^T$. Augmenting the system model (equation 2.12) with the measurement models used to drive it (equations 2.13 and 2.14), the estimate will evolve according to

$$\begin{bmatrix} \dot{\hat{x}} \\ \dot{\hat{y}} \\ \dot{\hat{\phi}} \\ \dot{\hat{b}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{\phi} \\ \hat{b} \end{bmatrix} + \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ \omega \end{bmatrix} + \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_u \\ v_v \\ v_\omega \\ v_b \end{bmatrix} \tag{2.15}$$

or, more succinctly,

$$\dot{\hat{x}} = A\hat{x} + B(x)u + C(x)v$$

and the associated covariance

$$\dot{P} = AP + PA^T + C(x)\Sigma C(x)^T \tag{2.16}$$

Parameter	Measurement/Parameter	Axis	Value	Units	Reference
σ_u	body-relative velocity	$^b x$	0.001	$m.s^{-1}$	[124]
σ_v	body-relative velocity	$^b y$	0.001	$m.s^{-1}$	[124]
σ_{ω_z}	angular velocity	$^b z$	0.1	$rad.s^{-1}$	[34]
σ_{b_z}	angular velocity bias	$^b z$	0.3-3	$mrad.s^{-1}$	[34]
σ_z	depth	$^w z$	0.25	m	[35]

Table 2.1: Standard deviation values for a typical navigation payload comprising a Doppler velocity log (DVL), an AHRS or tactical-grade IMU, and a pressure sensor. The standard deviation in the depth measurement assumes an accuracy of 0.25% of full scale (assumed to be 100m).

Note that the need to project the body velocity in the world frame (using the heading estimate $\hat{\phi}$) makes the model non-linear.

2.2.4 Map Accuracy

The position of the sonar measurement in the world frame is

$$^W x = {}^W x_P + {}^W_P R \left({}^P x_S + {}^P_S R \mathbf{b} r_m \right) \tag{2.17}$$

where $^W x_P$ and $^W_P R$ are the position and orientation of the platform in the world frame. The uncertainty associated with this range measurement, expressed in the world frame, can be obtained using the same approach as in subsection 2.2.2, yielding

$$^W \Sigma_p = {}^W \Sigma_t + J_A \left({}^W \Sigma_a \right) J_A^T + {}^P_S R \left({}^P \Sigma_p \right) {}^P_S R^T \tag{2.18}$$

where the $^W \Sigma_t$ and $^W \Sigma_a$ terms are the uncertainty in the position and attitude of the platform (respectively). The covariances for the noise models driving this equation are listed in table 2.1. The term $^P \Sigma_p$, capturing the uncertainty in the measurement in the platform frame, is defined by equation 2.8.

The discussion above describes a "decoupled" navigation filter, where estimates for the horizontal and vertical position components are obtained independently. This approach, while simple, provides a good approximation to the navigation schemes found in the literature and used by several underwater platforms [48, 49].

Chapter 3

Multibeam Data Processing

3.1 Introduction

The purpose of this chapter is to introduce the primary mapping sensor—the multibeam sonar—and the techniques used to process its data so as to derive measurements that can be spatially registered using the chosen map representation. To do this, we begin with an overview of the operating principles behind the sonar and how these are used to produce an image. We then state the segmentation problem, which must be solved in order to obtain a set of range measurements that can be spatially registered. Before moving on to the problem of segmentation, however, we discuss some important pre-processing techniques that can be used to mitigate noise and other artifacts present in these images that will otherwise affect the accuracy of the range estimates. We address the segmentation problem with two alternative approaches: the first estimates target and background intensity distributions to obtain a per-pixel (*dense*) segmentation; the other assumes a signal with a known envelope may exist in the received reflection and estimates its location to obtain a range estimate directly. Both techniques are extended to take advantage of underlying scene geometry. Finally, we demonstrate and compare the proposed approaches with the standard *fixed-threshold* segmentation technique using experimental data.

While the main focus of this chapter is on the processing and segmentation of multibeam profiling sonar data, the proposed approaches generalize well to other single- or multi-beam sensors which measure received reflection intensity as a function of time or range, with radar based systems being a prime example.

3.2   Related Work

The use of multibeam echo sounders for underwater mapping dates back a few decades [101]. Featureless sonar-based mapping techniques rely on classifying sonar images into "object" and "free-space" regions (*segmentation*) to derive estimates of range to objects in the scene, which are then spatially registered using the latest pose estimates. As described in the previous chapter, the accuracy of this range estimate depends not just on the accuracy of the sensor pose estimate, but also on the accuracy of the segmentation process.

Most featureless approaches rely on some form of threshold-based segmentation of the sonar image, selecting, for each beam, either the first or strongest return above a threshold intensity value. The use of a fixed threshold in the presence of scene-dependent intensities and sonar artifacts can often result in poor segmentation accuracy, as well as little robustness to noise or variation in the sonar image. To address these limitations, some techniques complement the threshold operation with additional image processing steps, such as morphological operations such as *dilation* and *erosion* [13]. Other approaches have also added some other form of post-segmentation outlier rejection, usually by looking at the agreement between the range measurements of neighboring beams to improve the accuracy of the range estimates [58, 61, 82]. Recognizing the difficulty in obtaining an accurate and robust segmentation using threshold-type methods, other approaches have instead addressed these limitations later in the pipeline [115]. This is achieved by employing outlier rejection and smoothing techniques on the point clouds that are obtained by spatially registering the range measurements. As a consequence, these techniques, while straightforward and robust to noise and other artifacts, may have limited ability to accurately reconstruct small-scale detail.

Many other sonar processing applications appear to take the opposite perspective, and point to more robust processing and segmentation techniques [14, 15, 63]. Early work on sonar data segmentation looked at using echo statistics for fish and seafloor classification [98, 99]. These methods were able to show that certain differences in the scene (e.g. the density of a fish school) resulted in significantly distinct echo statistics, and proposed the use of the Rayleigh and Rice distributions to model the different features. Other related work, performed in the context of submarine detection, addressed the problem of reducing the number of potential detections (*clutter removal*) in a low-frequency sonar ping [20, 53]. Intensity distributions were estimated using the sonar image in combination with the output from a Page detector [3]. The segmentation problem was then formulated as a Markov random field, which was solved using these distribution estimates and some manually set parameters. More recent work addressed the problem of obtaining these distribution estimates without the use of a detector, relying instead on *expectation-maximization* techniques

to determine the parameters of a mixture model [2].

3.3 Problem Statement

The problem we address in this chapter can be described as follows: we are interested in finding a function $f(\cdot)$ that operates on one or more sonar scan images Y, to produce a set of measurements (r_k, α_k) describing the range and azimuth (respectively) to a set of scatterers in the scene. Each pixel y_{ij} ($i \in \{1, ..., R\}$, $j \in \{1, ..., B\}$) in the sonar scan image is assumed to take a quantized intensity value in the set $\{0, ..., L - 1\}$ which may or may not be calibrated —i.e. the relationship between the image intensity y_{ij} and the corresponding acoustic intensity may be unknown. As this function will operate on the intensity measurements, it is also critical that these measurements are as accurate as possible. Ensuring this is the case requires identifying—and mitigating—the primary mechanisms through which inaccuracies are introduced in the sonar measurements.

3.4 Sonar Operating Principles

In order to devise techniques that estimate range measurements from sonar scans we must first understand how the image is formed, and what are the relevant acoustic phenomena affecting that process. We do so by first looking at the single-beam sonar through the sonar equation and understanding the significance of each of its terms.

3.4.1 The Sonar Equation

In its simplest form, a single-beam active sonar transmits an acoustic signal at some time t_0 and samples the received acoustic signal over some time interval $[t_0 + \delta, t_0 + \delta + RT_S]$. Here, δ denotes the blanking time (used to mitigate misdetections due to reverberation in the transducer), T_S is the sampling period, and R is the number of samples. Assuming no occlusions and a constant sound speed c, this interval corresponds to the range interval (or *window*)

$$[r_{min}, r_{max}] = \frac{c}{2}[t_0 + \delta, t_0 + \delta + RT_S] \tag{3.1}$$

with the kth sample of the received signal having been reflected by an object located at a range

$$r_k = \frac{c}{2}(\delta + kT_S) \tag{3.2}$$

or, equivalently, somewhere in the kth range bin. The received signal intensity RL for that sample can be modeled using the (active) *sonar equation* [12, 55, 113]:

$$RL = SL + DI - 2TL(r_k) + TS \tag{3.3}$$

Here, the source level SL is the intensity of the transmitted acoustic signal at the source, while the directivity index DI models the variation in that intensity with direction—also known as its beam pattern. Both parameters are specific to the actual sonar, with data sheets often reporting the beam width as the angle for which DI reaches $-3dB$ (*i.e.*, intensity is halved). The transmission loss $TL(r)$ models the attenuation of the signal as it travels between the sonar and the scatterer - it is a function of the distance between the two and it also depends on medium properties such as the spreading and absorption regimes under which the sonar operates. Finally, the target strength TS will depend on the the object's dimensions and scattering properties. All of these quantities are expressed in *decibels* with respect to a reference pressure - typically, $1\mu Pa$: $dB\ re\ 1\mu Pa$. For the sake of clarity, however, we will omit both the reference pressure ($1\mu Pa$) and distance ($1m$) and simply write "dB".

As the received signal is acquired from an acoustic transducer through conditioning, sampling, and analog-to-digital conversion, it will be quantized to values in the set $\{0, \ldots, L - 1\}$, where $L = 2^b - 1$ is the dynamic range of the acquisition front-end (typically $b = 8$ or 16 bits). Thus, for a single-beam sonar, the output will be a one-dimensional array of quantized intensity values indexed by range, $y(r_k)$.

3.4.2 Sonar Properties

Beampattern

One of the terms of the sonar equation introduced in the previous section (eq. 3.3) is the directivity index DI, which models the sonar's beam pattern, *i.e.*, its directional sensitivity. The beam width of a sonar, defined by the angle at which the main lobe is $3dB$ below its maximum value, determines its angular resolution. The side lobes determine other directions to which the sonar is highly sensitive—depending on the difference in amplitude between main and side lobes, reflections originating from these directions will be more or less noticeable, introducing inaccuracies in the measurements and (further) reducing the overall resolution of the sonar. The effect of these mechanisms is illustrated in Figure 3-3, where multiple, fainter instances of the hull shape are visible as a result of significant side lobe height.

3.4.3 Attenuation

The transmission loss term in equation 3.3, TL, captures the attenuation the acoustic signal suffers as it travels from the sonar to the scatterer and back. This reduction is due to two phenomena affecting the signal: *geometric spreading* over an increasingly large surface as it travels in a dispersive medium, and *absorption* of the signal by the medium as some of the energy in the pressure waves that carry it is converted to heat:

$$TL(r) = TL_S(r) + TL_A(r) \tag{3.4}$$

The behavior of these two terms is summarized in figure 3-1.

Geometric Spreading

Geometric spreading models the decrease in intensity as the signal wave front propagates away from the source along a surface that is increasing in area. Depending on the sonar and environment, this spreading is often modeled as *spherical* or *cylindrical*, with $1/r^2$ and $1/r$ dependencies, respectively. Given a reference distance of $r_{ref} = 1$, the transmission loss due to geometric spreading, in $dB.m^{-1}$, can be written as

$$TL_S(r) = 10\log_{10}\left(r^{\beta}\right) \tag{3.5}$$

The parameter β models the spreading regime: $\beta = 1$ for cylindrical, and $\beta = 2$ for spherical.

Absorption

As the signal travels through water, there will also be a decrease in its intensity caused by dissipation, as some of its energy is converted to heat. This dissipation is caused by viscous drag that the water particles are subject to as they move, and by molecular relaxation processes in some of the chemical components in sea water, namely boric acid and magnesium sulphate. These three mechanisms result in absorption losses that are proportional to range:

$$TL_A(r) = (\alpha_V + \alpha_B + \alpha_M)r \tag{3.6}$$
$$= \alpha r$$

where α represents the *total* absorption, caused by viscous (α_V) and molecular relaxation (α_B, α_M) processes. These are mainly dependent on sonar *depth* and *frequency*, as well as water *salinity* and *temperature*, and models for these are available in the literature [113, Ch. 5], [12, Ch. 5].

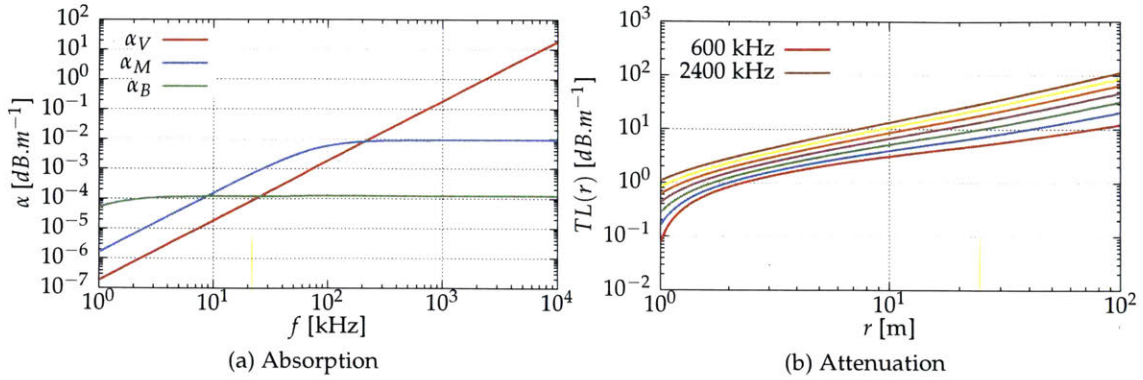(a) Absorption   (b) Attenuation

Figure 3-1: Absorption (left) and attenuation (right) curves under cylindrical spreading ($\beta = 1$) for the frequencies and ranges typical of most multibeam sonars, in 300kHz increments [12, 113].

3.4.4   Speed of Sound

Equation 3.2 assumed a constant speed of sound, c. This assumption is important for several reasons: first, if the speed of sound is constant in the region of interest of the surrounding medium, then there will be no refraction of sound rays, and these can be treated as straight lines in space. Second, it allows for the relation between the delay in the received echo and the range to the scatterer that originated to be written as the linear relationship expressed by equation 3.2, as opposed to a more complicated form.

While assuming a constant speed of sound certainly simplifies the process of range estimation, it does not eliminate all of the underlying uncertainty; letting $t = 0$ and $t = \tau$ denote the times at which the pulse was sent and echo was received, respectively, equation 3.2 can be simplified to:

$$\hat{r} = \frac{1}{2}\hat{c}\hat{\tau} \tag{3.7}$$

where we have used $\hat{\cdot}$ to denote an estimate. From this we can see that our range estimate is highly sensitive to errors in speed of sound and timing estimates. To first order, these can be approximated as:

$$\hat{r} \approx \frac{1}{2}(c + \delta_c)(\tau + \delta_\tau)$$
$$\approx r + \frac{1}{2}c\delta_\tau + \frac{1}{2}\tau\delta_c \tag{3.8}$$
$$\approx r + \delta_r$$

Here, c and τ represent the true values of the speed of sound and signal delay, respectively. The uncertainty in δ_τ is dominated by the uncertainty in the method used to estimate τ. For the method described later in the text (section 3.6.2), this is proportional to $\frac{c}{B}$ [45], but

without much loss of generality we can assume the order of magnitude for δ_τ to be $\mathcal{O}(10\mu s)$ or smaller. The uncertainty in the speed of sound estimate, in turn, can range from as little as $10ppm$ (for speed of sound measurements obtained from dedicated sensors [62]) to $15ppt$ for fixed estimates, if the latter are assumed to deviate as much as $20m.s^{-1}$ from a nominal value of $1500m.s^{-1}$ (for typical seawater conditions). A very common approach, found in many underwater platforms, is to estimate the speed of sound from on-board pressure and temperature sensors using a well-established model [29].

3.4.5 Multibeam Sonars

A multibeam sonar, comprising an array of transducers, uses beamforming to create a set of virtual beams spanning a sector of a plane, each of which can be thought of as a single-beam sonar as described above[1]. Figure 3-2 illustrates the scanning geometry and output of a multibeam sonar: a two-dimensional array (image), representing quantized intensity measurements over a polar grid, with each column corresponding to a virtual beam. A sonar with B beams and R range bins (or equivalently, samples over time) will output an $Y_{R \times B}$ array, where $y_{ij} \in \{0, \ldots, L-1\}$ represents the quantized average intensity of the received signal for the jth beam's i^{th} range bin.

3.5 Pre-processing

The previous section described some of the most important phenomena affecting sonar measurements; in this section we introduce some of the techniques that can be employed to mitigate their effects.

3.5.1 Beampattern

In most multibeam sonars the effects of the sonar's beam pattern manifest themselves as artifacts in the scan images—some of these are illustrated in figure 3-3. In order to mitigate these effects, we can model the sonar as a linear, spatially invariant system $h(r, \alpha)$ under the effect of additive noise v:

$$\tilde{y}(r, \alpha) = h(r, \alpha) * y(r, \alpha) + v \tag{3.9}$$

---

[1]Some multibeam sonars employ acoustic lens assemblies to accomplish beamforming [6], while some others make use of a 2D array to sweep the beam in two directions and produce a volumetric scan. For simplicity, and without loss of generality, we focus our discussion on typical imaging and bathymetric multibeam echo sounding sonars, comprising a one-dimensional linear transducer array.
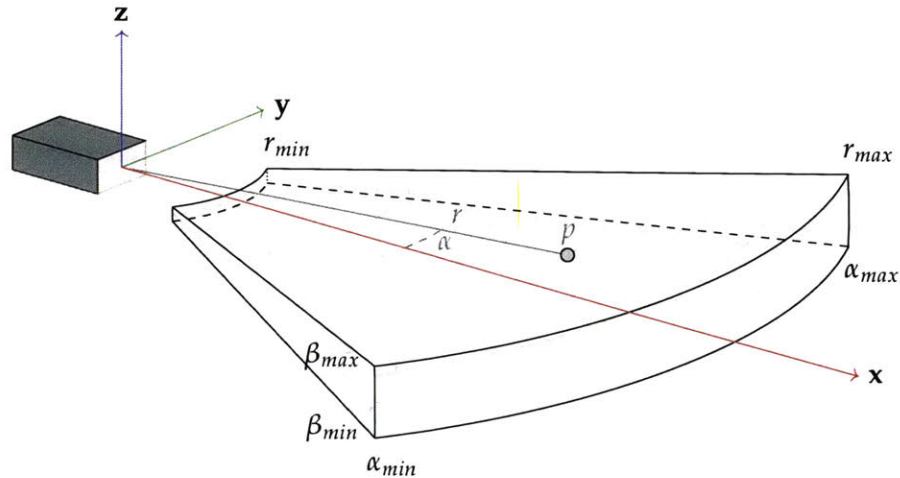
Figure 3-2: Reference frame and scanning geometry for a multibeam echo sounder. The horizontal gridded plane (gray, $\beta = 0$) indicates the projection of the image plane: element i, j in the image corresponds to the point r_i, α_j in the scanning plane (i.e. to the ith range bin in beam j). For a *profiling* multibeam sonar, the nominal field of view in elevation (β) is much wider than in azimuth (α).

(a)   (b)   (c)   (d)
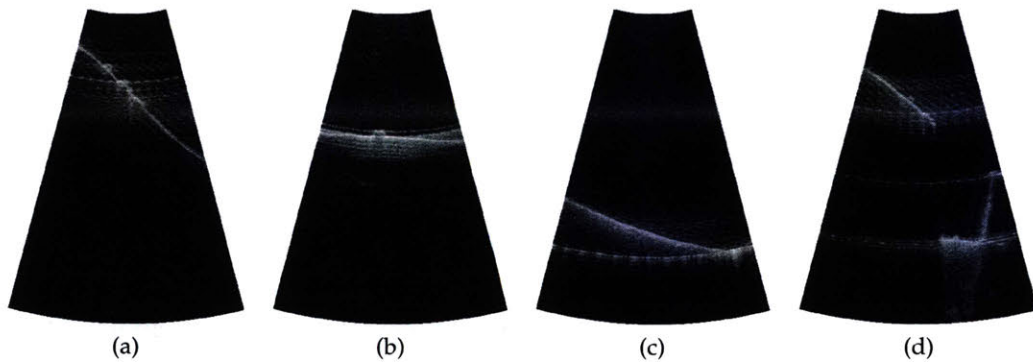
Figure 3-3: Sample scans from a *profiling* multibeam sonar. The artifacts and outliers present in these scans are caused by a combination of acoustic phenomena that are governed by the properties of the sonar, medium, and scattering object(s). Techniques to mitigate these effects are described in section 3.5.

If we decouple this model into its angular and radial components, and consider only the former $(h(r, \alpha) = h(\alpha))$, the impulse response corresponds to the sonar's beam pattern. Using this model, we can employ Wiener deconvolution to recover an estimate of the original image y from the noisy image $\tilde{y}$ [38]:

$$\hat{x}(r, \alpha) = g(r, \alpha) * y(r, \alpha)$$
$$G(\omega_\alpha) = \frac{H^*(\omega_\alpha)}{|H(\omega_\alpha)|^2 + SNR(\omega_\alpha)^{-1}} \tag{3.10}$$

$H(\omega_\alpha)$ and $SNR(\omega_\alpha)$ are, respectively, the power spectral densities of the impulse response and signal to noise ratio.

It is important to note that the spatial invariance assumption on $h(\alpha)$ is clearly invalid: this amounts to requiring the beam pattern—the location, width and magnitude of lobes and nulls— to be the same for all azimuths. While this is rarely—if ever—the case, it nonetheless constitutes a useful approximation, provided that the actual beam pattern does not deviate significantly over the chosen approximation over the azimuths of interest, as will be demonstrated later. Moreover, we can at still mitigate part of the variation in the beam pattern shape (the taper in lobe height with the variation in azimuth) by introducing an angle-varying gain $k(\alpha)$.

3.5.2 Attenuation

The attenuation effects described in subsection 3.4.3 are often corrected with a time-varying gain (TVG) on the sonar, or as a radial gain on the image [82]:

$$k(r) = k_1 \log_{10}(r) + k_2 r + k_3 \tag{3.11}$$

where, from equations 3.5 and 3.6, we would set $k_1 = 20\beta$ and $k_2 = 2\alpha$.

3.6 Scan Segmentation

Having described some of the most important phenomena affecting sonar data and the techniques used to mitigate their effects, we now turn to the problem of classifying those data to obtain the measurements required to build a model of the surrounding scene.

We propose two different types of segmentation techniques, *dense* and *sparse*, according to the level of detail with which they segment data: the former classify every site (*pixel*) in a sonar scan, while the latter classify individual beams only. In both types of techniques, we will limit the scope of our segmentation to binary classification; in other words, we are simply concerned about the existence (or absence) of a scatterer in a given *pixel* or *beam*.

$$x_{ij}$$

$$y_{ij}$$

Figure 3-4: Dense MRF model for a single site (i, j) - y_{ij} is the measurement (quantized acoustic intensity at r_i, α_j), and x_{ij} is the unknown state (e.g. "background" and "target" for $x_{ij} = 0, 1$, respectively). A sonar scan can be treated as $R \times B$ of these models, assumed independent.

By contrast, more advanced techniques could, for instance, distinguish between different *types* of scatterers according to their acoustic properties (e.g. wood, metal, rock, sand, ...).

3.6.1   Dense Segmentation

Dense segmentation techniques aim at estimating a label for every *pixel* in an incoming sonar scan. Assuming most of the sonar and channel effects have been dealt with through adequate pre-processing, the problem of dense segmentation can be thought of as that of assigning a label to each pixel while considering its intensity to be a noisy measurement of the target strength term in equation 3.3.

More formally, the *dense* segmentation problem can be described as that of estimating the label x_{ij} from a measurement y_{ij} - this can be seen as a *Hidden Markov Model*, where the latent variable x_{ij} is either "empty" (0) or "occupied" (1), and the observation y_{ij} is a quantized intensity level, as illustrated in figure 3-4. Under this assumption of independent sites, the joint probability for a set of assignments $x = \{x_{ij}\}$ is given by:

$$p(x) = \prod_{i,j} p(x_{ij}|y_{ij}) \tag{3.12}$$

Thresholding

One of the segmentation techniques most commonly found in the literature is that of (fixed) *thresholding* [58, 61, 82]. In this type of approaches, the intensity of each pixel is compared with an intensity threshold ϵ, and a scatterer is assumed to be present at the corresponding site ($x_{ij} = 1$) if $y_{ij} \geq \epsilon$; otherwise the site is considered empty ($x_{ij} = 0$):

$$y \underset{x=0}{\overset{x=1}{\gtrless}} \epsilon \tag{3.13}$$

The most significant limitation of this approach is that its performance is highly dependent on the scene: different scenes may have very dissimilar objects in them, with highly contrasting acoustical properties - a particular threshold value is likely to be adequate for one but not the other. Moreover, even given the same scene and type of object (a ship hull, for instance), there will likely be a dependence on how the object is ensonified, as for many scatteres the reflected intensity will vary with the incidence (grazing) angle [118].

Segmentation approaches based on fixed intensity thresholds often have to face a decision between a "low" threshold, that correctly identifies most objects while also selecting many outliers, and a "high" threshold, with improved outlier rejection but which often fails to detect objects. In some of the proposed approaches, these limitations are addressed through use of image processing techniques (such as morphological image operations [13]), or handled later in the mapping pipeline through the use of outlier rejection schemes at the map representation level [116].

For the reasons listed above, the value of ϵ should change with the scene—while operator adjustments are certainly possible, it would be preferable to do so based on a statistical model describing the intensity distribution for each object type in the scene, particularly for fully autonomous (non-supervised) platforms, as opportunities for operator intervention are limited or non-existent.

Intensity Distribution Estimation

Treating each site x_{ij} as independent of all other sites (figure 3-4), we can write the probability distribution associated with its measurement as an N-component mixture model

$$p_{ij}(y) = \sum_{k=1}^{N} \delta_{ij}(k) p_{ij}(y|k) \qquad (3.14)$$

where $\delta_{ij}(k) \in \{0,1\}$ and $\sum_{k=1}^{N} \delta_{ij}(k) = 1$ for all sites (i,j)—i.e., $\delta_{ij}(k)$ is the indicator variable for $x_{ij} = k$. It is important to note that the independence assumption made above is admittedly incorrect: at the very least, the intensity measurement y_{ij} will depend on whether or not a scatter is present in the preceding sites along the same beam, i.e., in sites $(1, \ldots, i-1, j)$. Still, this assumption is used to significantly simplify our approach, and is commonly made further along in a mapping pipeline [21]. Again, we have also implicitly assumed that transmission loss, directivity, and all other effects unrelated to the presence (and type) of scattering occurring at site (i,j) have been modeled and corrected for, so that $p_{ij}(y)$ depends mostly—if not only—on the scene.

Under this formulation, the conditional intensity distributions $p_{ij,k}(y|k)$ are free to have different parameters for different sites, radically increasing the dimensionality of the esti-

mation problem, even for single-parameter distributions. To address this, we make the additional assumption that these distributions are identical for all sites:

$$p_{ij}(y) = \sum_{k=1}^{N} \delta_{ij}(k) p(y|k) \qquad (3.15)$$

This is equivalent to treating all the *pixels* in a sonar scan as realizations of the same mixture model, albeit with different prior (π_k) values.

At this point, the conditional distributions are still unknown; to estimate them, we can average equation 3.15 over all sites (i, j):

$$\begin{aligned}
p(y) &= \frac{1}{BR} \sum_{i,j} p_{ij}(y) \\
&= \frac{1}{BR} \sum_{i,j} \left(\sum_{k=1}^{N} \delta_{ij}(k) p(y|k) \right) \\
&= \sum_{k=1}^{N} p(y|k) \left(\sum_{i,j} \frac{1}{BR} \delta_{ij}(k) \right) \\
&= \sum_{k=1}^{N} \pi_k p(y|k)
\end{aligned} \qquad (3.16)$$

where $\pi_k \in [0, 1]$ are the N component priors, with $\sum_{k=1}^{N} \pi_k = 1$. If $p(y|k)$ are assumed to follow parametric distributions, the estimation problem can be framed and solved using techniques such as expectation-maximization or least-squares, to obtain estimates for both π_k and the component parameters.

Mixture Components   The received signal intensity will depend greatly on the type and quantity of objects present in the scene: the scattering properties of a metal hull are markedly different from those of a sandy seafloor, and even within the same class we should expect some variability. For these reasons, the component choices will be tied to the application and/or scene. Some of the options found in the literature include [2, 99]:

- the exponential distribution, with *pdf* [30, p. 88-92],

$$p_Y(y) = \frac{1}{\beta} e^{-\frac{y}{\beta}} \qquad (3.17)$$

- the Rayleigh distribution, with *pdf* [30, p. 173-175]:

$$p_Y(y) = \frac{y}{\sigma^2} e^{-\frac{1}{2}\left(\frac{y}{\sigma}\right)^2} \qquad (3.18)$$

Both of these distributions have the non-negative real half-space ($\mathbb{R}_0^+$) for support. As previously mentioned, the intensity measurements output by a sonar are quantized to 2^b bits, so it is also important to consider the discrete analogues of these distributions:

- the *geometric* distribution, with *pmf* [42, p. 210] :

$$p_I(i) = p(1-p)^{i-1} \tag{3.19}$$

- the *discrete Rayleigh* distribution, with *pmf* [86]:

$$p_I(i) = \theta^{i^2} - \theta^{(i+1)^2} \tag{3.20}$$

with θ parameterized as $e^{-\lambda}$.

While both of these distributions are defined over the infinite set $i \in \mathbb{N} \cup 0$, in nearly all cases, the total probability outside the interval of interest $(0, \ldots, L-1)$ will be negligible:

$$\sum_{i=0}^{+\infty} p_I(i) - \sum_{i=0}^{L-1} p_I(i) \approx 0$$

Independent of the parametric models chosen for each mixture component, the distribution described by equation 3.16 should not be treated as stationary (except, perhaps, for static scenes): as the scene changes, so will the parameters, and the mixture model must be re-estimated for every new image. This phenomenon is illustrated later in this chapter (figure 3-17) using a short segment from a ship hull inspection session. Plotting the intensity histograms for each of the sonar scans we notice a significant variation in the distribution of intensity values beyond the "baseline" (background) low intensities. This variation will be analyzed in greater detail in section 3.7.

The MAP classifier

In the special case where $N = 2$ (e.g. "empty" and "occupied" for $k = 0, 1$, as before), eq. 3.16 becomes

$$p(y) = \pi_0 p(y|x = 0) + (1 - \pi_1)p(y|x = 1) \tag{3.21}$$

Under this model, the binary *maximum a-posteriori (MAP)* (MAP) decision rule is [69]:

$$\Lambda(y) \underset{x=0}{\overset{x=1}{\gtrless}} \eta \tag{3.22}$$

where $\Lambda(y) = \frac{p(y|x=1)}{p(y|x=0)}$ is the likelihood function, and $\eta = \frac{\pi_0}{1-\pi_0}$ is the prior ratio. In this binary hypothesis setting, the probabilities of *false alarm*, $p_{FA}(y)$ and detection, $p_D(y)$ are

also of interest:

$$p_{FA}(y) = 1 - P(y|x = 0) \tag{3.23}$$

$$p_D(y) = 1 - P(y|x = 1) \tag{3.24}$$

where $P(y|x = l)$ is the conditional cumulative distribution function for $x = l$. Plotting $p_D(y)$ against $p_{FA}(y)$ yields the *Receiver Operating Characteristic (ROC)* curve, which helps better understand classifier performance and the available trade-off between the two. The *area under curve (AUC)* is an important performance metric when comparing different classifiers:

$$AUC = \int_y \frac{p_D(y)}{p_{FA}(y)} dy \tag{3.25}$$

Finally, it is also worth noticing that these probabilities are often assumed known in some mapping pipelines, particularly those modeling the map as an occupancy grid [21, 36], so estimating these values from scene data should result in more accurate maps.

Dense Markov Random Field

The MAP classifier described above makes only use of local information, that is, it relies only on y_{ij} to estimate x_{ij}. Given that sonar scans will be spatially consistent and smooth (i.e., if a site is empty, it is highly likely that the neighboring sites are too), it would be beneficial to take advantage of the intensities in the neighborhood of (i, j) to estimate x_{ij}. This approach is illustrated in figure 3-5, where the 4-neighborhood of (i, j), $\mathcal{N}_4(i, j)$, is considered when computing x_{ij}. This *lattice* model is also known as a *Markov Random Field (MRF)* or *Ising* model, and it corresponds to the factorization

$$p(x_{ij}, y_{ij}) = \frac{1}{Z} \prod_{ij} p(x_{ij}, y_{ij}) \prod_{kl \in \mathcal{N}(ij)} p(x_{ij}, x_{kl}) \tag{3.26}$$

where $\mathcal{N}(i, j)$ is the set of indices (k, l) comprising the neighborhood of (i, j). Note that, with respect to eq. 3.12, we have now included the joint label distribution $p(x_{ij}, x_{kl})$ in our model.

Taking the logarithm of the joint probability $p(x_{ij}, y_{ij})$ yields the *joint likelihood*, $E(x_{ij}, y_{ij})$:

$$E(x_{ij}, y_{ij}) = -ln(Z) + \sum_{ij} \phi(x_{ij}, y_{ij}) + \sum_{kl \in v_{ij}} \psi(x_{ij}, x_{kl}) \tag{3.27}$$

The terms $\phi(x_{ij}, y_{ij})$ and $\psi(x_{ij}, x_{kl})$ are also known as the *unary* and *binary* potential functions (respectively). The unary potential function corresponds to the mixture model described above (eq. 3.16); the binary factor must either be constructed from labeled data, or
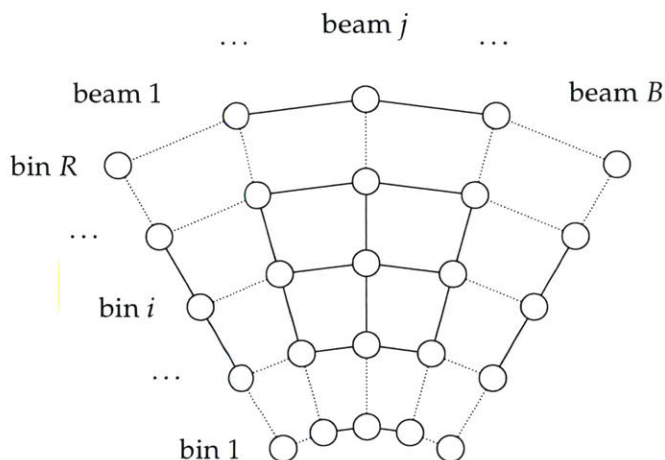
Figure 3-5: Graphical model for the dense Markov Random Field segmentation model - the measurement priors (previously shown in figure 3-4) have been omitted for the sake of clarity.

prescribed based on heuristics or prior knowledge of the scene.

Binary potential functions and neighborhood set   In the derivation above we have used $\mathcal{N}_{ij}$ to denote the neighborhood of site (i, j), that is, the sites whose label influence the label assigned to x_{ij}. The definition of this set, together with the potential function $\psi(x_{ij}, x_{kl})$ will determine the spatial smoothness of the solution: a large $\mathcal{N}_{ij}$ in conjunction with a "narrow" $\psi(x_{ij}, x_{kl})$ (i.e. heavily penalizing deviations between x_{ij} and x_{kl}) will produce a spatially smooth solution, likely at the expense of accuracy. Conversely, a smaller neighborhood with a "wide" binary potential function will yield results close to those of the local MAP classifier. Typical choices for $\mathcal{N}_{ij}$ are the $4-$ and $8-$ neighborhoods of a site; in the binary classification setting, $\phi(x_{ij}, x_{kl})$ is often chosen to be an indicator function:

$$\delta(x_{ij}, x_{kl}) = \begin{cases} 1, & \text{if } x_{ij} = x_{kl} \\ 0, & \text{otherwise} \end{cases} \tag{3.28}$$

Given the non-equal spacing between sites, this function can be assigned a weight to reflect the relatively lower importance of more distant sites, for instance:

$$w(x_{ij}, x_{kl}) = \|\mathbf{x}_{ij} - \mathbf{x}_{kl}\|^{-1} \tag{3.29}$$

where $\mathbf{x}_{ij} = [r_i \cos(\alpha_j)\ r_i \sin(\alpha_j)]^T$ is the position of site (i, j) in the sonar's reference frame.

Solution Techniques   The maximum *a posteriori* estimate of the label assignment $\mathbf{x} = \{x_{ij}\}$ is

$$
\begin{aligned}
x_{ij}^{MAP} &= \arg\max_{x_{ij}} p(x_{ij}, y_{ij}) \\
&= \arg\max_{x_{ij}} E(x_{ij}, y_{ij})
\end{aligned}
\tag{3.30}
$$

Due to the large size of the solution space ($2^{R \times B}$), the computational cost associated with finding the globally optimal solution to this optimization problem is prohibitive– to avoid this fact, some of the most popular techniques seek instead a local maximum. Informally, *Simulated Annealing (SA)* employs a Monte Carlo approach to randomly "walk" the solution space starting from some initial solution. Candidate solutions are evaluated and accepted depending on their cost and the number of iterations that has taken place. *Iterative Conditional Modes (ICM)*, on the other hand, chooses the optimal label for each site given the current assignment before recomputing the total energy. It then repeats the process until convergence or a maximum number of iterations has taken place [20, 53].

3.6.2   Sparse Segmentation

The dense segmentation techniques described above aim at classifying each site in the sonar scan, resulting in a segmentation problem with a very large number of variables: $R \times B$ sites. This may be considered inefficient given the goal of recovering at most one range per beam, particularly for sonars with a high bin count. At the same time, dense techniques do not properly model the occupancy of a sonar beam: ignoring acoustic effects such as reverberation and multi-path propagation, we should expect at most a single return per beam. This would mean that if $\hat{x}_{ij} = 1$, then $\hat{x}_{kj} = 0$ for all $k \in \{1, 2, \ldots, i - 1\}$, that is, all the preceding sites should be considered empty.

To address these limitations, we now propose an alternative formulation of the segmentation problem: we consider the hidden variable to be the position of the scatterer along the main axis of a beam, $r \in [r_{min}, r_{max}]$, and the observed variable $\mathbf{y}$ to be the received acoustic intensity over that interval, an R-dimensional vector $\mathbf{y}$ of (quantized) intensity values. Ideally, given $\mathbf{y}$, we would like to recover the belief over the range r to the scattering object or surface, $p(r|\mathbf{y})$, for each beam in the scan. The graphical model corresponding to this approach is illustrated in figure 3-6. Under this model, the joint probability for a set of range measurements $\mathbf{r}$ is given by

$$
p(\mathbf{r}) = \prod_i p(r_i | \mathbf{y_i})
\tag{3.31}
$$

r_i

y_i

Figure 3-6: Sparse MRF model for a single beam: y_i is the measurement (quantized acoustic intensity for the ith beam, over the range $[r_{min}, r_{max}]$), and r_i is the unknown state - the range to the scatterer. A sonar scan can be considered as up to B of these models.

The MAP classifier

If the conditional distribution $p(r|\mathbf{y})$ is known, then the MAP estimate for the model shown in figure 3-6 is

$$\hat{r} = \arg\max_r (p(r|\mathbf{y})) \tag{3.32}$$

Recovery of $p(r|\mathbf{y})$ requires an understanding of a sonar's operating principles that is different from the one previously described, where each site was associated with an independent measurement of RL (eq. 3.3); now, we must instead understand how that measurement varies over time (or, equivalently, range).

The Matched Filter

In an idealized detection problem, a sonar transmits a known signal $u(t)$, which is then reflected back towards the sonar by an object at a range r_0. In this scenario, determining the range to the scattering object amounts to detecting the presence and location of the transmitted signal $u(t)$ in the noisy received signal:

$$y(t) = u(t - \tau) + w(t) \tag{3.33}$$

where $\tau = 2\frac{r_0}{c}$ is the unknown two-way travel time, and $w(t)$ is often assumed to be additive white Gaussian noise (AWGN) with variance σ^2.

Under these assumptions—additive noise and linear medium response—the optimal estimator, known as the *matched* filter [92, 117], is obtained by correlating the known signal $u(t)$ with the received signal $y(t)$:

$$
\begin{aligned}
q(t) &= u(t) \star y(t) \\
&= \frac{2}{N_0} \int_{t_{min}}^{t_{max}} y(\tau) u(\tau - t) d\tau
\end{aligned}
\tag{3.34}
$$

where N_0 is the energy in the noise signal:

$$N_0 = \int_{-\infty}^{+\infty} |w(t)|^2 dt$$
$$= \frac{1}{2}\sigma^2$$

(3.35)

Provided the assumptions made in subsection 3.4.4 hold, a change of variable $r = \frac{c\tau}{2}$ lets us rewrite the correlation above as a function of range:

$$q(r) = u(r) \star y(r)$$
$$= \frac{4}{N_0 c} \int_{t_{min}}^{t_{max}} y(s)u(s-r)ds$$

(3.36)

The conditional distribution $p(r|\mathbf{y})$ can now be recovered through the exponential mapping [123]:

$$p(r) = \exp q(r)$$

(3.37)

$$p(r|\mathbf{y}) = \frac{1}{Z}p(r)\exp\left(q(r)\right)$$

(3.38)

where Z is a normalizing constant.

An example of the matched filtering process is illustrated in figure 3-7, where a windowed sinusoidal signal $u(t)$ is transmitted by the sonar and reflected by an object at an unknown range $r = \frac{c\tau}{2}$: $p(t|\mathbf{y})$ is recovered from the correlation $q(t)$ of the received signal $y(t)$ with the transmitted signal $u(t)$.

If the prior belief on r is uniform, then the *maximum a posteriori* (MAP) range estimate, $\hat{r}_0$, is simply the value of r at which $q(r)$ is maximum:

$$p(r_i) = \arg\max_r \exp q(r)$$
$$= \arg\max_r q(r)$$

(3.39)

The width of the peaks in the correlator output—and thus the resolution of the range estimate—is dependent on the properties of the transmitted signal. The autocorrelation function for a single-frequency pulse, for instance, will have a much wider peak than a linear frequency modulated (*chirped*) pulse. Consequently, the resolution of the resulting range estimate will be significantly poorer. This dependence is captured in the expression describing the variance associated with the MAP estimate of r_i [55, p. 215]:

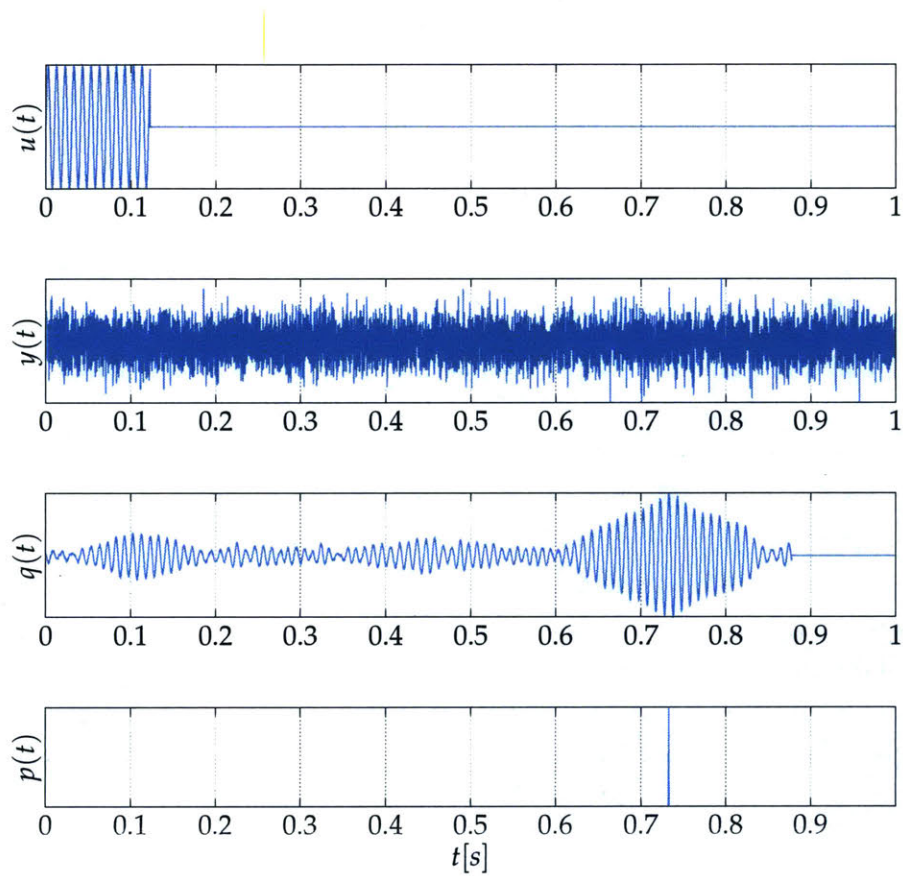$$\sigma_r = \frac{B}{2E(2\pi B_e)^2}$$

(3.40)

Figure 3-7: Matched filter example (after Woodward [123, Ch. 5]): the transmitted pulse $u(t)$ is reflected by an object and present in the received signal $y(t) = u(t - \tau) + n(t)$. The correlation of the transmitted and received signals $q(t)$ is then used to estimate a belief $p(t)$ over the location of the reflected signal.
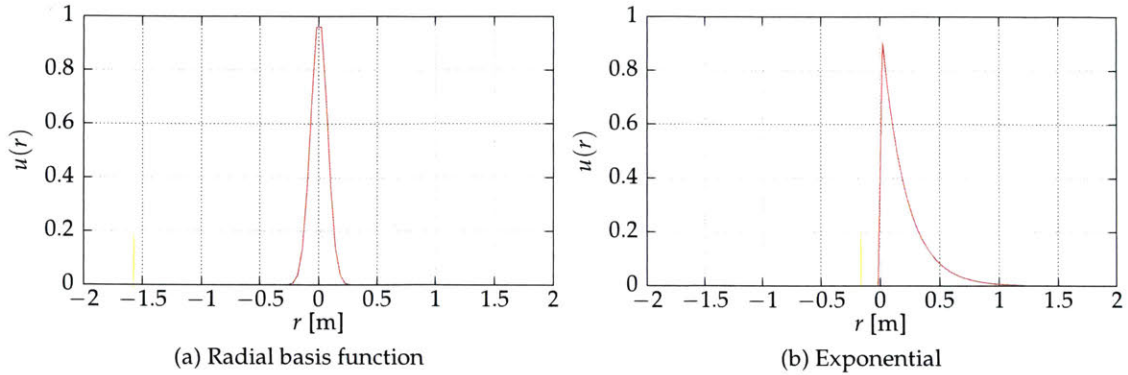
(a) Radial basis function   (b) Exponential

Figure 3-8: Pulse envelope candidates: RBF (left) and exponential (right) pulses.

where B is the received signal bandwidth, E and B_e are the energy and *effective* bandwidth (respectively) of the transmitted signal.

Just as most multibeam sonars do not provide direct control over the properties of the transmitted signal (other than its center frequency), it is often the case that they do not also provide direct access to the received signal (let alone to the correlator output); instead, what is typically output to the user is the uncalibrated acoustic intensity—in other words, the energy of the received signal. While this clearly does not meet the matched filter assumptions, we can nevertheless apply the same principle of using the correlation with the transmitted signal as a proxy for the belief over target location over time (or range), with the understanding that the guarantees offered by the standard matched filter are likely to no longer hold. To do so, we must first choose some approximation for the envelope of the known transmitted signal. From experimental observation of multibeam sonar data, two reasonable options would be *Gaussian* and *exponential* pulses, shown in figure 3-8.

"Empty" beams   So far in the discussion we have not yet considered the case in which a beam does not ensonify a scatterer - the "empty" beam scenario. In this situation the model described above would have to be extended - one possibility would be to add a dedicated label or state, such as $r = \emptyset$. To avoid the edge case associated with this label when defining a transition function, we instead opt for skipping the beam altogether when no reflection is deemed present. While this constitutes a second detection problem, this proves to be much simpler, as we are concerned only with the *presence* of a scatterer along a beam (and *not* its location), for which the total received energy can prove to be an accurate indicator. Thus, the first step in this approach is to determine whether an object has been ensonified by a beam before using the approach described by eq. 3.36 to determine its

location.

Sparse Markov Random Field

Having described a *per-beam* MAP classifier, we can now take a similar step to the one taken to extend the dense segmentation approach: consider information in neighboring sites to improve performance. Assuming a spatially smooth scene (such as a ship hull), it is highly likely that the range measurement for beams $i - 1$ and $i + 1$ will not be too different from that of beam i. This model is illustrated in figure 3-9, where the 2-neighborhood of i, $\mathcal{N}_2(i)$ is considered when estimating r_i. This model (it too a *Markov Random Field*) is also known as a *Potts* model, and it corresponds to the factorization:

$$p(\mathbf{r}) = \prod p(r_i) \prod_{j \in \mathcal{N}(i)} p(r_i, r_j) \tag{3.41}$$

where $\mathcal{N}(i)$ is the set of indices comprising the neighborhood of i, and $p(r_i, r_j)$ is the joint distribution. As before, taking the logarithm yields

$$E(r_i, y_i) = -\log(Z) + \sum_i \phi(r_i, y_i) + \sum_{j \in \mathcal{N}} \psi(r_i, r_j) \tag{3.42}$$

where the terms $\phi(r_i, y_i)$ and $\psi(r_i, r_j)$ are the *unary* and *binary* potential functions. The first term corresponds to the empirical distribution derived from the matched filter model described above.
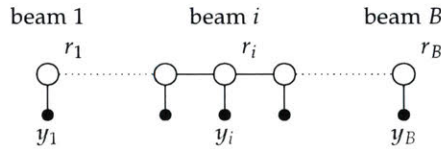

Figure 3-9: Sparse MRF model.

Transition function   The term $p(r_i, r_j)$ in eq. 3.41, also known as the *transition function*, is heavily dependent on both the scene geometry and the sensor pose. To better illustrate this dependence, consider the top half of figure 3-10, where the sonar ensonifies a flat wall from two distinct viewpoints; each of the viewpoints will result in a different transition function, as shown on the lower half of the figure. While this dependence on both the scene and pose knowledge will be addressed more explicitly in chapter 5, for now it is important to highlight that the choice for the transition function is tied to implicit assumptions regarding both scene geometry and sensor viewpoint, requiring a trade-off between

(a) Normal incidence scene

(b) Oblique incidence scene

(c) Normal incidence: transition function

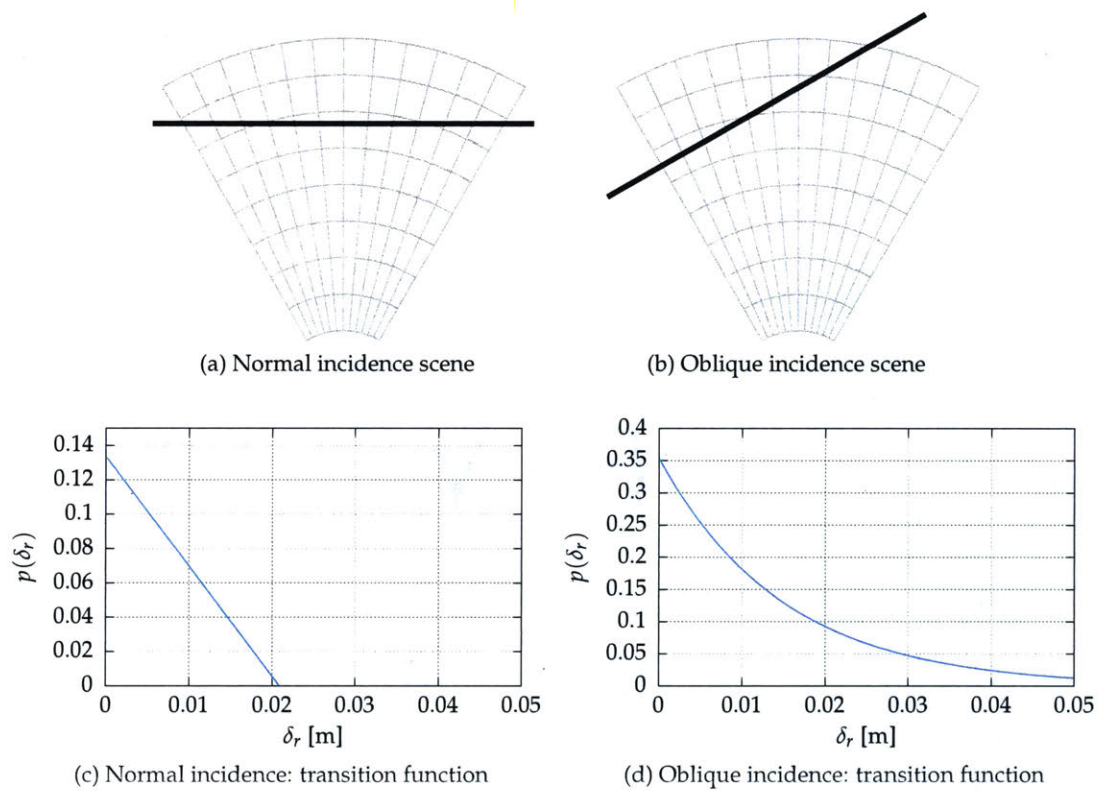(d) Oblique incidence: transition function

Figure 3-10: Transition function dependence on scene geometry and sensor pose for the sparse MRF approach: the same scene under two different viewpoints (top) can result in significantly different (empirical) transition likelihood functions (bottom). This scenario assumes a sonar with 512 equi-spaced beams covering a field of view of 60° imaging perpendicularly (left) or at an angle (right).

performance and robustness: just as strong assumptions on $p(r_i, r_j)$ will yield poor performance in areas where these assumptions do not hold, relaxing these too much will offer little benefit over the local MAP classifier (eq. 3.36) at the expense of added computation.
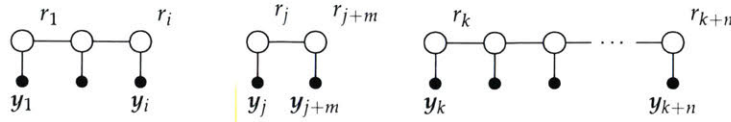

Figure 3-11: Sparse MRF segmentation with subgraphs - each connected component (subgraph) is solved as an independent problem.

Solution Techniques   This sparse formulation of the segmentation problem can be solved using the same techniques as those used for the dense approach - SA and ICM. However, due to the (comparatively) much lower dimensionality of the problem space, direct computation of globally optimal solutions is feasible.

3.6.3   Multi-scan Segmentation

Both the *dense* and *sparse* techniques described in the previous subsections aim at classifying individual sonar scans. However, just as the performance of local models for each of the two approaches could be improved by considering information in neighboring sites within the same scan, one could argue for further extending the proposed techniques by considering neighboring sites in adjacent scans as well. This concept is illustrated in figures 3-12 and 3-13, where the models from figures 3-5 and 3-9 were extended to include information from the scans at $t-1$ and $t+1$, becoming three- and two- dimensional *lattices*, respectively. Under this model, eqs. 3.26 and 3.41 would remain the same, except for the definition of $\mathcal{N}(x_{ij})$ and $\mathcal{N}(r_i)$ (respectively), which would now extend both within and *across* sonar scans. The main challenges with this approach revolve around the fact that defining the neighborhoods would require knowledge (explicit or assumed) about both sensor pose and scene geometry. As the sensor moves in the world, it will not always be the case that, say, x_{ij}^t is adjacent (or in the neighborhood of) x_{ij}^{t-1} and/or x_{ij}^{t+1}. Similarly for sparse techniques, the neighborhood of a site r_i^t will depend on both its current estimate and the sonar pose. For these reasons, we will skip these extensions for now and revisit the concept in chapter 5.
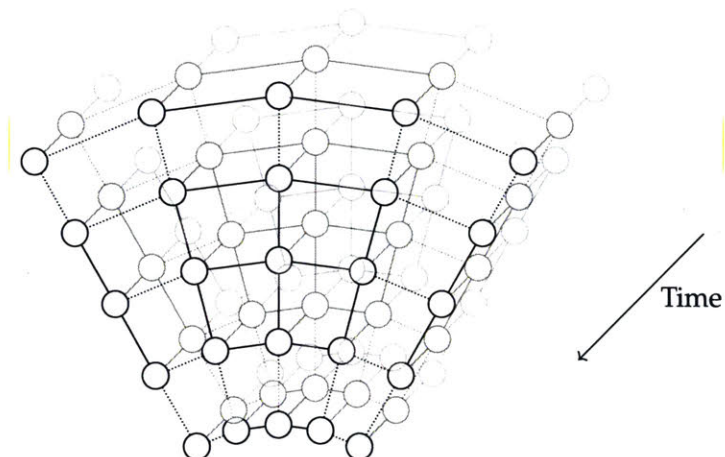
Figure 3-12: Temporally-extended dense MRF segmentation model. Each of the nodes in the graph represents a site $x_{i,j,k}$, where the indices i, j, and k index the beam, bin, and scan. The measurement priors $y_{i,j,k}$ have been suppressed for the sake of clarity.
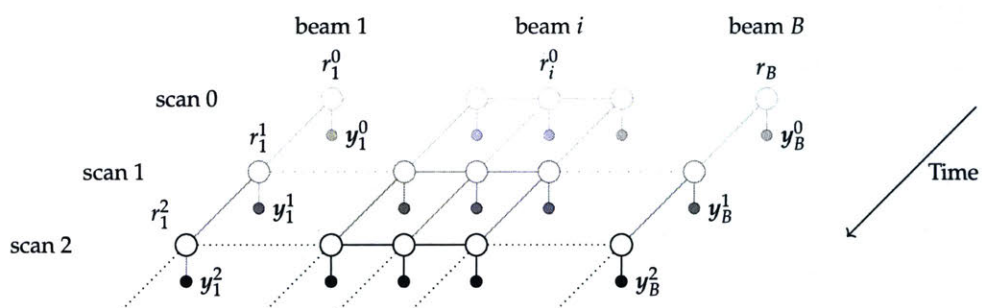

Figure 3-13: Temporally-extended sparse MRF model.

3.7 Experimental Results

In this section we present experimental results for the methods described in the previous sections. We begin by demonstrating the pre-processing techniques described in section 3.5 on real data from a multibeam sonar. This pre-processed data is then used to derive statistical intensity models like those described in section 3.6.1, and to demonstrate and evaluate the segmentation algorithms proposed in sections 3.6.1 and 3.6.2.

3.7.1 DIDSON test set

Sonar

The *Dual-frequency Identification Sonar (DIDSON)* [6, 19], is a multibeam sonar designed for imaging applications, often being described as an "acoustic camera". It can operate at either 1.1 or 1.8 MHz center frequencies, corresponding to its *detection* and *identification* modes. Unlike many multibeam sonars, the DIDSON uses an acoustic lens assembly to accomplish beamforming, achieving a nominal beamwidth of 0.3° in azimuth and 1.0°, 14.0°, or 28.0° in elevation, depending on the lens used. For profiling applications, a *concentrator* lens is used, resulting in the 0.3 × 1.0° nominal beamwidth. DIDSON scans comprise 96 beams and 512 range bins, and are thus represented as 8-bit 512 × 96 arrays.

Test set

The scans in the test set were selected in order to represent a diverse set of scenes, namely: propeller, rudder, sea floor, and different hull geometries. Taken from a larger inspection data set, covering the aft section of a ship (running gear), this set comprises 3000 scans acquired over approximately 7 minutes. All scans were acquired with the DIDSON configured in high frequency (1.8 MHz) mode, with a 9 meter window starting from 2.25 meters.

Pre-processing Results

In section 3.5 we described the sonar's beampattern as one of the phenomena behind the artifacts in sonar scans. While the DIDSON does not accomplish beamforming through typical means, it is still subject to these effects, as shown in figure 3-14. Additionally, the interleaved fashion through which the sonar forms a scan, together with the individual beampatterns, make these artifacts particularly noticeable: the sonar fires one set of 12 transducers at a time, repeating the process over a total of 8 cycles to form a full image; as the mainlobe of a transducer is nearly aligned with the sidelobes of other transducers

firing in the same cycle, the received signal will be "contaminated" by other signals in the same cycle. This results in the "tunnelling" artifact that can be seen in many profiling scans, and can often prove challenging for the extraction of range measurements [115].

It is important to note that the interleaved scan forming technique has another important consequence: as all the pixels in the scan are not acquired simultaneously, sonar data will be sensitive to motion, particularly rotations along the scanning plane. These effects, while definitely important, are deemed to be specific to this particular type of sonar (uncommon in profiling applications) and are not addressed in this work.

The results of these pre-processing techniques are shown in figures 3-25b through 3-30b. When compared with the original scans (figures 3-25a through 3-30a), there is a noticeable improvement in both the angular and radial directions, clearly visible in scans with flat scenes (36, 1096, 1578, and 2981): both the "tunneling" artifacts, and the angular drop-off in intensity caused by the beam-pattern characteristics are significantly less pronounced. As those scans show, this also improves the radial resolution of the scan - not only are the surfaces sharper, but it is also easier to identify objects attached to the hull, such as those in scans 36 and 338.

### Intensity Distribution Estimation Results

Evaluation   To evaluate the fit between the empirical intensity distribution and the proposed mixture model we use the *Kullback-Leibler divergence (KLD)*, $D(p,q)$, which measures the *divergence* between a "true" distribution p and a *model q* [2]

$$
\begin{aligned}
D(p,q) &= -\sum_i p(i) \log \frac{q(i)}{p(i)} \\
&= \sum_i p(i) \log p(i) - \sum_i p(i) \log q(i) \\
&= -H(p) + H(p,q)
\end{aligned}
\tag{3.43}
$$

where $H(p)$ and $H(p,q)$ are the *entropy* and *cross-entropy*, respectively.

Background data set   In order to inform the component choices for the mixture model, we begin by selecting a subset of scans in the test set for which no objects are imaged. By looking at the intensity distributions for the scans in this set we hope to gain some insight regarding the appropriate distribution to employ, as well as the typical parameter values. Figure 3-16 shows the intensity histogram for this "background" set of scans. Ignoring (for now) the value at $y_i = 0$, we see from the constant slope in the logarithmic plot that this

---

[2]Informally, KLD can be considered the information lost in approximating p with q.
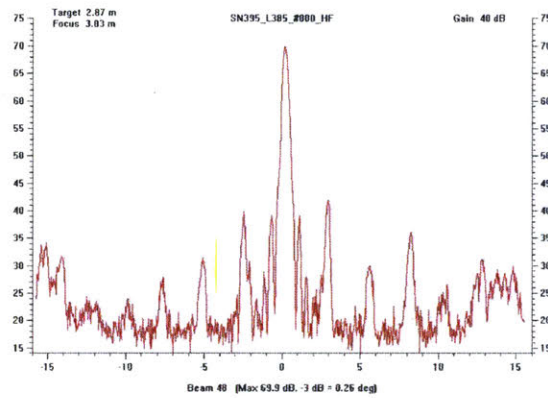
(a) Beam pattern for the center beam
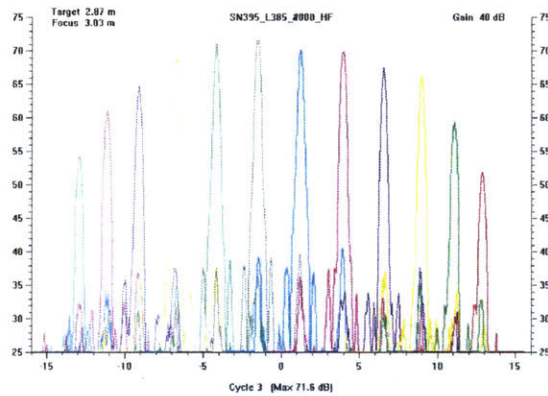
(b) Beam patterns in the 4th firing cycle
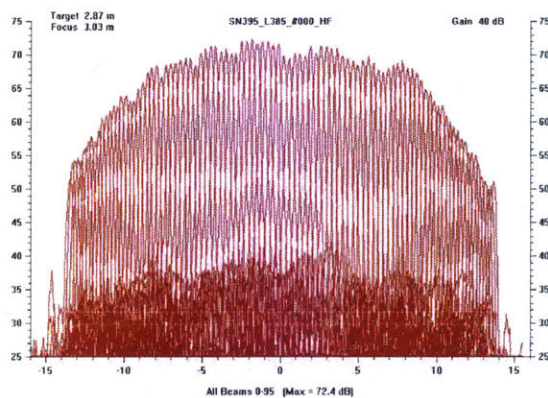
(c) All beam patterns

Figure 3-14: DIDSON: beam patterns for center beam (a), beams in the 4th firing cycle (b), and all beams (c). While the overall beampattern shape remains approximately the same for all transducers, there is a noticeable taper away from broadside ($\alpha = 0°$) [Figures courtesy of Sound Metrics Corp.].
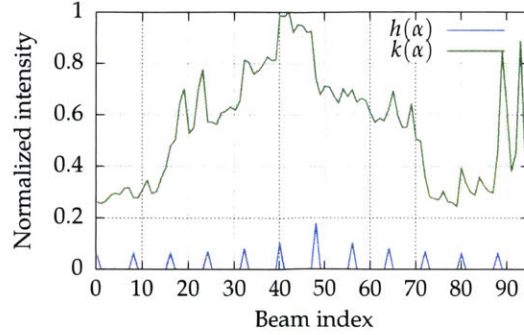
Figure 3-15: Angular (azimuth) impulse response $h(\alpha)$ and taper function (α) for the DID-SON. These functions were obtained from a combination of the sonar specification (figure 3-14) and experimental data. The impulse response function was normalized to have unit energy, and the taper function was normalized to unit gain at $\alpha = 0°$.

empirical distribution can be well approximated using an exponential distribution. For the exponential distribution parameterization described by equation 3.17, the unbiased *MLE* of the β parameter is

$$\hat{\beta} = \frac{1}{N}\sum_i y_i \tag{3.44}$$

Using the non-zero values of y_i to estimate β we obtain $\mu(\beta) = 21.15 \times 10^{-3}$ and $\sigma(\beta) = 0.972 \times 10^{-3}$. Setting $\hat{\beta} = \mu(\beta)$ yields $D(p,q) = 11.0 \times 10^{-3}$ (computed all over non-zero y_i values in the "background" set).

Zero-bias   To finalize our choice for a "background" intensity distribution component we must address the histogram value at $y = 0$. To account for this bias, we modify the exponential model above to increase the probability at $y = 0$:

$$p_Y(y|x = 0) = (1 - \pi)\delta(y) + \frac{\pi}{\beta}e^{-\frac{1}{\beta}} \tag{3.45}$$

where we use the empirical distribution value at zero, $h(0)$, to set the weight of the exponential component:

$$\pi = \frac{1 - h(0)}{1 - \beta^{-1}}$$

This bias is likely the result of a combination of limitations in both the sensitivity of the individual transducers used in the sonar and the dynamic range of the sonar's acquisition front-end (signals below a certain amplitude threshold will register as zero amplitude).
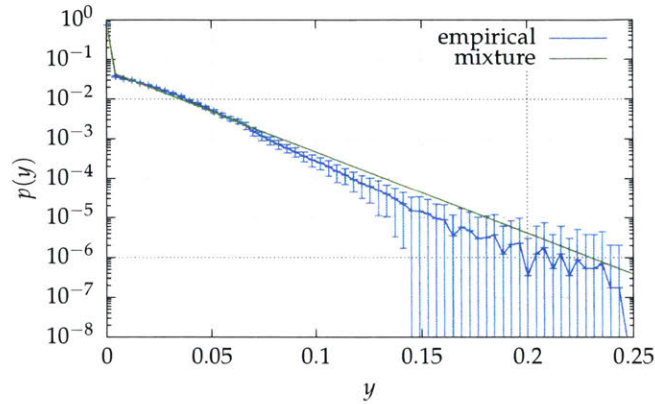
(a) Empirical distribution and mixture model
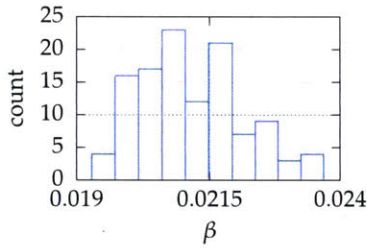
(b) Exponential: scale parameter
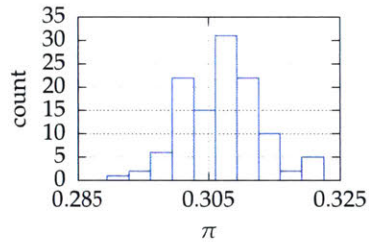
(c) Exponential: weight
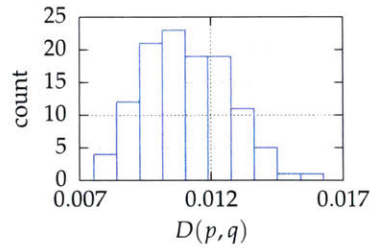
(d) KL divergence

Figure 3-16: Background intensity distribution model: (a) aggregate intensity histogram and mixture model, (b) weight (π) histogram, (c) scale parameter (β) histogram, and (d) KL divergence histogram. This background model was obtained from a subset of 116 scans where no objects were present in the sonar field of view. Sample mean and standard deviation for π and β are $0.307 \pm 5.90 \times 10^{-3}$ and $0.0212 \pm 0.972 \times 10^{-3}$, respectively, and $0.011 \pm 1.6 \times 10^{-3}$ for $D(p,q)$.

Mixture Model   Assuming that each site (i,j) can have one of two labels, $x_{ij} = 0$ for "background"/"empty" and $x_{ij} = 1$ for "object", we write (equation 3.21):

$$p(y) = \pi_0 p(y|x=0) + (1 - \pi_1)p(y|x=1)$$

Replacing $p(y|x=0)$ with the model described above, and assuming object intensity to follow a Rayleigh distribution (eq. 3.18), the mixture model can be rewritten as:

$$p_y(y) = (1 - \pi_1' - \pi_2')\delta(y) + \pi_1'\frac{1}{\beta}e^{-\frac{y}{\beta}} + \pi_2'\frac{y}{\sigma^2}e^{-\frac{y^2}{2\sigma^2}} \tag{3.46}$$

where π_1' and π_2' are the weights of the exponential and Rayleigh components.

Figure 3-19 shows the mixture model results for sample scans in the ship hull and sea wall data sets—from these we can see that the mixture model accurately approximates the empirical distribution for all of the chosen sample scans. The model parameters, listed in table 3.1, show that the KL divergence between the two is less than 5×10^{-3} for all samples. The performance over the entire dataset, shown in figures 3-18 and 3-19, is, for the most part, similar: divergence remains below 1×10^{-2}, and the model parameters—scales and weights—are consistent throughout the scan. Still, it is important to highlight a few noteworthy aspects:

- It is possible to distinguish, from either the empirical or model distributions, different parts of the scene: rudder, propeller, and hull. The segment contains three lawn-mower legs where each leg is performed at constant depth. Legs 1 and 3 move from aft of the rudder towards the front of the ship; leg 2 returns to the aft of the rudder.

- While both rudder and hull have similar intensity distributions, and can be distinguished by the discontinuities in figure 3-19, scans where the propeller is ensonified result in a significantly different estimate for σ - this is caused by the strikingly distinct geometry of the propeller, which results in a much larger number of sites attaining higher intensity values.

- Mixture model approximation accuracy is notably poorer in segments where the sonar does not ensonify an object. These result in larger divergence values (close to 1×10^{-2}), as well as poor estimates for the Rayleigh scale parameter, σ. In these situations, the Rayleigh component of the mixture model is unnecessary, so its scale parameter is undefined.
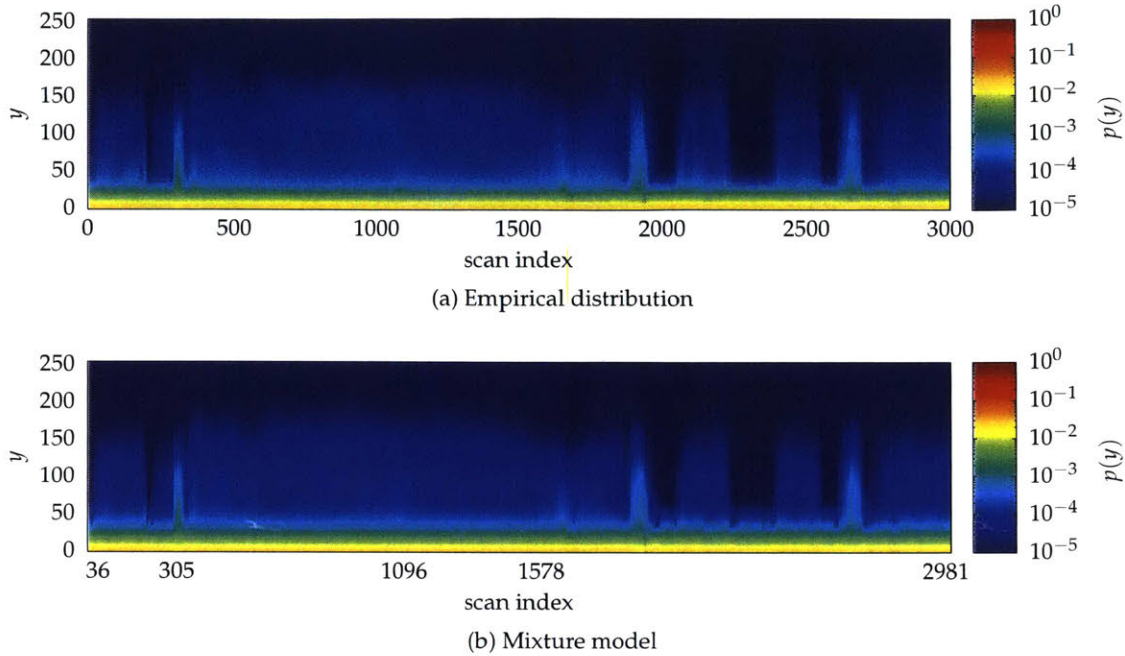
(a) Empirical distribution

(b) Mixture model

Figure 3-17: DIDSON ship hull data set - empirical distribution (top) and mixture model (bottom). These plots illustrate the change in the empirical intensity distribution throughout the 3000-scan segment. The segment begins aft of the rudder, moving forward through the rudder, propeller, and hull; around scan 1100 the vehicle reverses direction and begins the return leg. The segment terminates slightly forward of the propeller, with the vehicle moving forward on a third inspection leg that began around scan 2350. The bottom plot illustrates how the mixture accurately approximates the empirical intensity distribution (consistent with the results in figure 3-18d), providing some experimental support to our choice of mixture components. In both plots we can distinguish some of the major ship features, namely: the rudder (scans 0-200, 2100-2200, 2400-2500), the propeller (around 300, 1950, and 2250), and the hull (scans 320-1800 and 2700-3000). The propeller, in particular, is quite distinct, due to the much higher likelihood of higher intensities caused by its geometry. The location of the sample scans (figures 3-25 through 3-30 are annotated on the horizontal axis of the bottom plot.)

Scan	Description	π_1	$\beta\,[\times 10^{-3}]$	$\pi_2\,[\times 10^{-3}]$	σ	$D(p,q)\,[\times 10^{-3}]$	AUC
36	rudder, anodes	0.328	32.534	10.582	0.244	3.997	0.995
305	propeller	0.285	36.688	72.770	0.191	2.870	0.991
338	shaft	0.326	32.122	7.662	0.273	3.847	0.996
1096	hull	0.318	31.849	13.988	0.260	3.279	0.996
1578	hull (inlet)	0.333	30.991	12.101	0.241	4.810	0.995
2981	hull (bottom)	0.332	30.417	8.656	0.268	4.656	0.996

Table 3.1: Mixture model parameters for sample scans from the DIDSON ship hull data set, shown in figures 3-25 through 3-30. The mixture models and the empirical distributions they approximate are plotted in figure 3-19.

(a) Component weights

(b) Component parameter - background (exponential) scale factor

(c) Component parameter - object (Rayleigh) scale factor

(d) KL divergence (KLD)

(e) Area under curve (AUC)
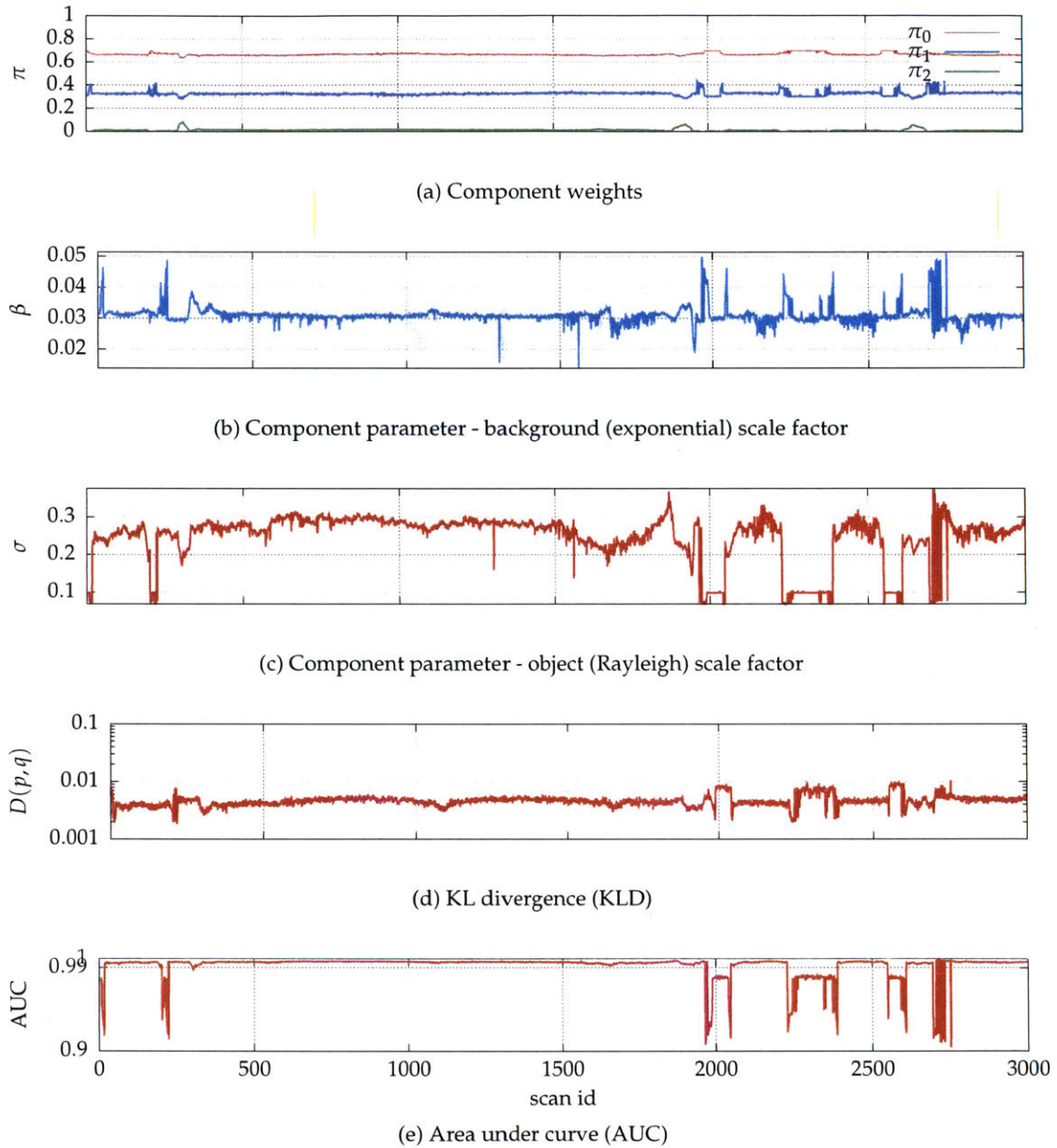
Figure 3-18: DIDSON ship hull data set - mixture model parameters for the 3000-scan segment. Both component parameters (scale factors β and σ) and weights (π_0, π_1, and π_2) remain relatively consistent throughout the segment, with a few notable exceptions, as can be seen from (d). These happen mostly in and around the interval between scans 2000 and 2500, on parts of the inspection segment where the sonar ensonifies empty space. Here we would expect $\pi_2 = 0$, and σ to be undefined.

Figure 3-19: Empirical intensity distributions and approximating mixture models for the sample DIDSON test scans, shown in figures 3-25 through 3-30. T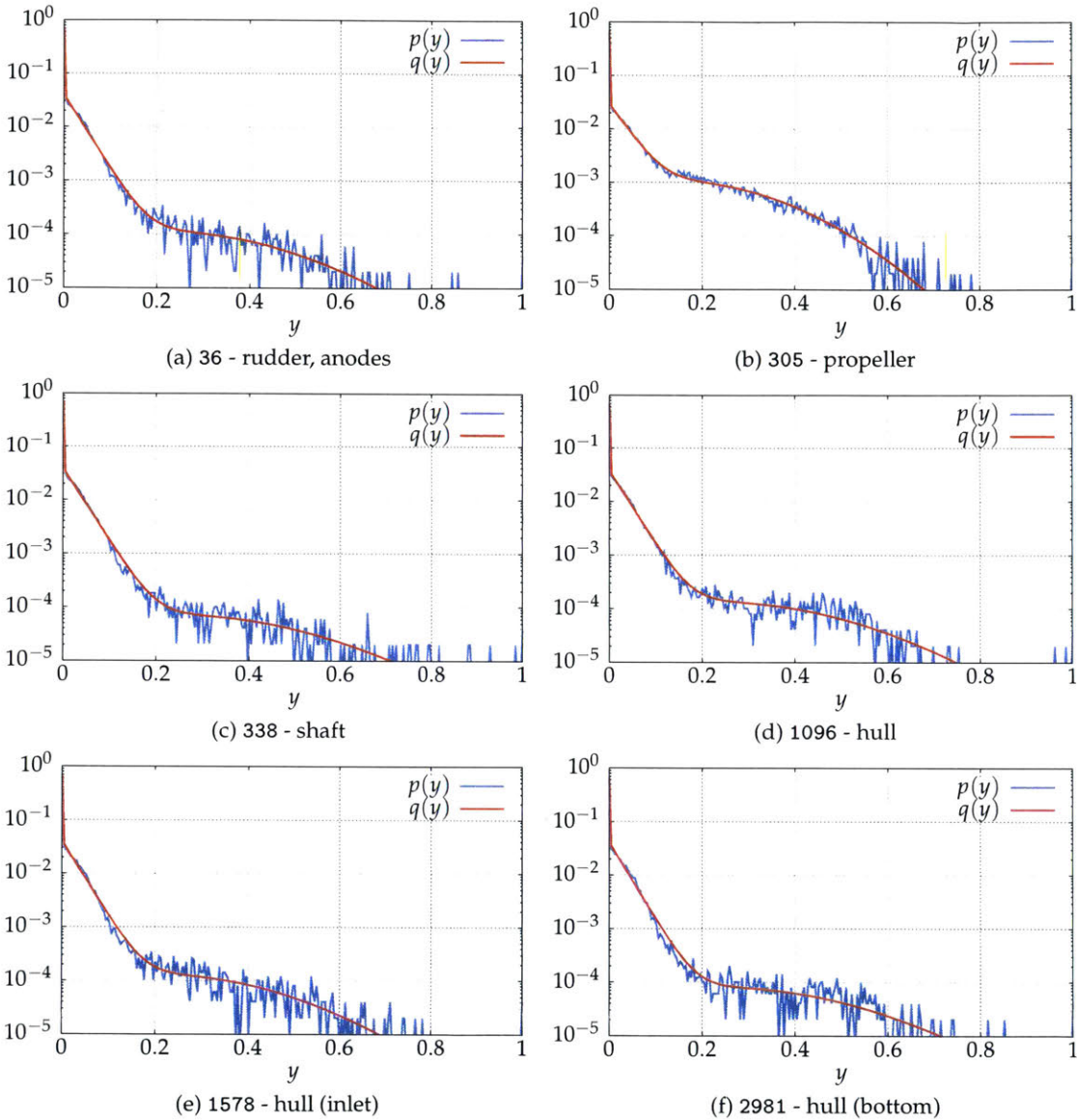he mixture model parameters are listed in table 3.1. All six plots show great agreement between empirical and model distributions, with the same overall shape: a bias at $y = 0$, followed by an exponential decay in intensity—typical of empty space in the scan. The occlusion of the field of view by an object manifests itself through the "knee-point" in the curve: high intensities are, for the most part, very unlikely, as the object appears as a single line in the scan. Still, there are some exceptions created by more complex scene geometries, as illustrated in figure (b) - here, the complex shape of the propeller results in a much higher number of sites attaining high intensity values, and the comparatively higher weight of the object component in the mixture model—nearly one order of magnitude.
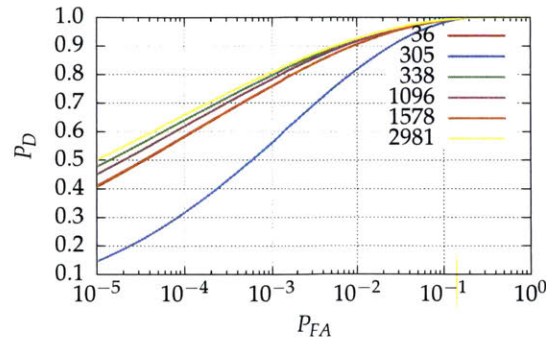
Figure 3-20: Receiver Operating Characteristic (ROC) curves for the DIDSON test scans. These were computed using the mixture models estimated using the techniques described in section 3.6.1. The segmentation performance is consistent, with the notable exception of scan 305. This scan, partially ensonifying the ship's propeller, yields an empirical distribution that is significantly different from those of the other test scans, with much higher likelihoods for high intensities. This is very likely caused by the richer scene geometry captured in a single scan, which is visibly different from that found on rest of the ship. For the other scans, the detection probability at $p_{FA} = 10^{-5}$ (approximately one false positive per scan) is about 0.5 - only half of occupied sites are likely to be detected.

Segmentation Results

Dense Segmentation   Using the mixture models discussed above we can quickly obtain the local (per-site) MAP segmentation of each scan (eq. 3.22). Local MAP segmentation results for each of the sample scans are shown in figures 3-25c through 3-30c. With the exception of scan 305, the AUC values for these scans, listed in table 3.1, range between 0.995 and 0.996. The ROC curves, shown in figure 3-20, show that the MAP segmentation achieves $p_D >= 0.9$ at $p_{FA} \approx 1 \times 10^{-2}$. Qualitatively, the segmentation correctly identifies the scene components in all six scans while also resulting in some outliers (most notably in scans 305 and 338) - these are the result of a combination of noise in the actual scan and artifacts that have not been fully mitigated in the pre-processing step. The comparatively worse segmentation performance achieved in scan 305, as seen in both the MAP segmentation (figure 3-26) and the ROC curve (figure 3-20), is likely the result of the limitations in the mixture model approximation previously discussed. The chosen components, while accurate at approximating the intensity distribution for most of the hull, do not perform as well in scans ensonifying more complex geometries such as those of a propeller. These limitations can also be seen in figure 3-18e: the AUC value, which exceeds 0.99 for most scans, drops to the range 0.9-0.99 for certain segments, namely those covering empty areas, such as the gap between propeller and rudder, and aft of the rudder. The MRF-based segmentation, whose results are shown in figures 3-25d through 3-30d, ameliorates some of the
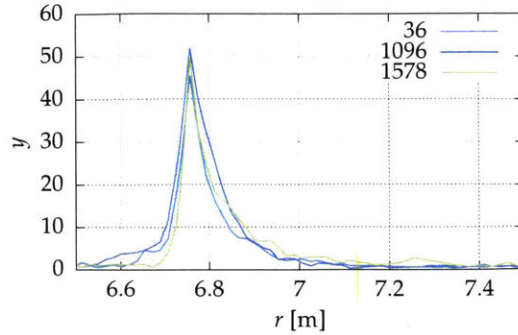
Figure 3-21: Average pulse shape for scans 36, 1096, and 1578. These pulses were computed by first aligning the signals in all beams to the same range value and then taking their average.

issues described above: there is a noticeable reduction in the amount of noise and outliers present in all scans. These segmentation results were obtained by iterative maximization of

$$X^\star = \arg\max_{x_{ij}} (1 - w) \sum \phi + w \sum_{ij} \psi \tag{3.47}$$

where the parameter w controls the influence of the binary factors (i.e. the smoothness of the final solution). The neighborhood of each site, $\mathcal{N}(x_{ij})$, was chosen to be the set of 8 immediately adjacent sites, each carrying an equal weight. The binary potential was simply the indicator function:

$$\psi(x_{ij}, x_{kl}) = \delta(x_{ij} - x_{kl})$$
$$= \begin{cases} 1, & \text{if } x_{ij} = x_{kl} \\ 0, & \text{otherwise} \end{cases} \tag{3.48}$$

To determine the label assignment from equation 3.47 we computed, at each step and for each site, the energy associated with each label (assuming all other labels remained the same). At the end of each step the label with maximum energy is chosen for each site, and the procedure is repeated until convergence (no change in the assignment X) or a maximum number of iterations has been reached.

Sparse Segmentation In order to use the sparse segmentation techniques described in subsection 3.6.2 we must first choose a pulse template that we can match against the received intensity for each beam. Figure 3-21 shows empirical approximations of the received pulses for scans 36, 1096 and 1578—these have been reconstructed from the pre-processed scans by aligning the received signals in every scan by their maxima. Given

these results, from the two pulse templates shown in figure 3-8, the exponential envelope appears to provide the best approximation. Having chosen a pulse template, the MAP solution can be determined for each beam from the correlation between the template and the received pulse—equation 3.41. We should note, however, that this equation only addresses the *location* of the pulse in the received signal, assuming its existence; by taking the correlation maximum of every beam we risk picking up range measurements from noise in otherwise "empty" beams. To address this fact, we require a minimum correlation value to derive a range measurement:

$$r_i = \begin{cases} \arg\max_r q_i(r), & \text{if } \max q_i(r) > \epsilon \\ \emptyset, & \text{otherwise} \end{cases} \tag{3.49}$$

The segmentation results for the hull data set are shown in figures 3-25e through 3-30e, where we have used the exponential pulse template previously described and a correlation threshold of $\epsilon = 0.4$. The local MAP technique is able to produce accurate range measurements with relatively few outliers, with the exception of scan 338—here we find outliers on the empty part of the scene. This is likely a consequence of choosing a correlation threshold ϵ that is potentially too low. Conversely, in scans 36 and 338, while the measurements accurately follow the rudder and hull shapes (respectively), they fail to pick-up the (fainter but valid) echo from the protruding objects (anodes). The consequences of choosing the strongest results are also visible in scan 1578, where the range measurements identify the back-plate of an inlet on the hull whereas the dense approach identified both the inlet and the back-plate.

The improvements brought about by MRF-based segmentation (figures 3-25f through 3-30f) are mostly visible on the smoother segmentation output (scans 36, 305, and 1096) and reduced number of outliers (scans 338, 1578, and 2981). These improvements are made possible by the MRF's ability to eliminate gaps between nearby beams: in this case, we have set $\mathcal{N}(r_i)$ to comprise its two adjacent beams. In order to determine the transition function to be used in the binary constraints we took the range measurements from the sparse MAP classifier above for the full 3000-scan segment and plotted the absolute difference histogram between valid, adjacent range measurements. As figure 3-23 shows, the most frequent absolute range differences between adjacent sites are well within half a meter, which is consistent with our smoothness assumption. While larger range differences are present, they are comparatively unlikely ($p < 1 \times 10^{-4}$ for $\delta_r > 2\ m$), so we (again) choose an exponential distribution as the transition function, where the scale parameter describes the smoothness of the solution (e.g. a large value for λ means that large range differences are extremely unlikely). Obtaining the label assignment using the MRF model
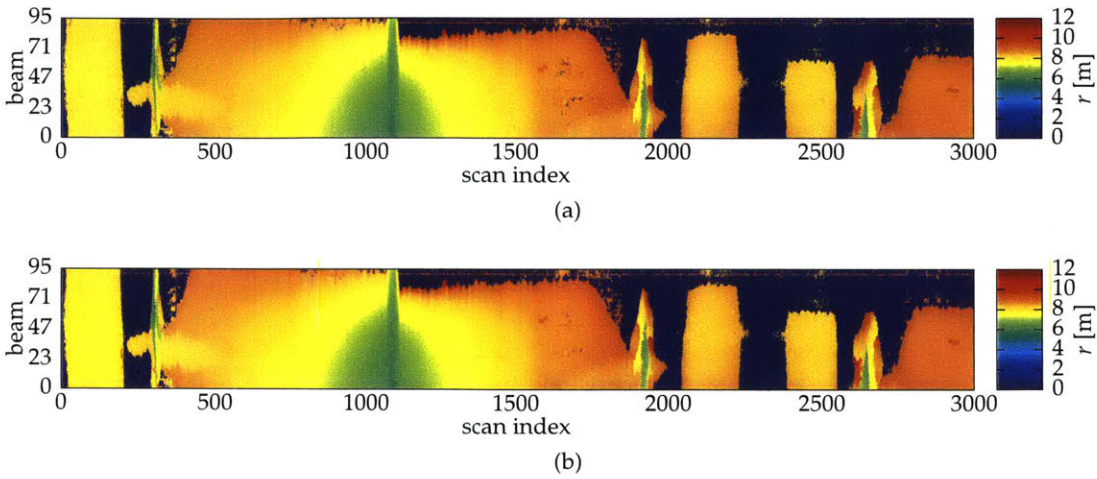
Figure 3-22: Range images for the sparse MAP (top) and MRF (bottom) classifiers. Deep blue denotes no range measurement ("empty" beam). The differences between the two, while subtle, is noticeable on the hull—between scans 1500 and 1750 the MRF results in much smaller number of gaps in its measurements (appearing as blue dots). Some of the smaller scale features visible in these images include the inlet forward of the propeller, near the bottom of the hull (towards the top of the image). From these images it is also possible to qualitatively recover the vehicle trajectory during the segment - three legs of a lawnmower pattern in increasing depth.

is achieved using the same technique as in the dense approach - iteratively solving for the assignment that maximizes the objective function by determining the optimal label for each site assuming all others as constant.

We also extend this solution technique to consider a varying transition function - as the number of iterations increases, so does the scale factor λ (illustrated in figure 3-24), rendering large discontinuities less likely.
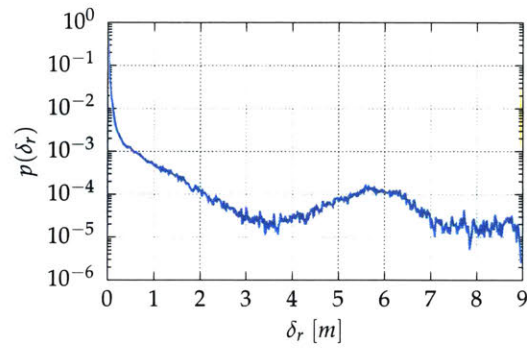
Figure 3-23: Empirical transition function obtained for the DIDSON hull data set: this function was estimated from the relative frequency of the absolute difference in adjacent range measurements obtained using the sparse MAP classifier.
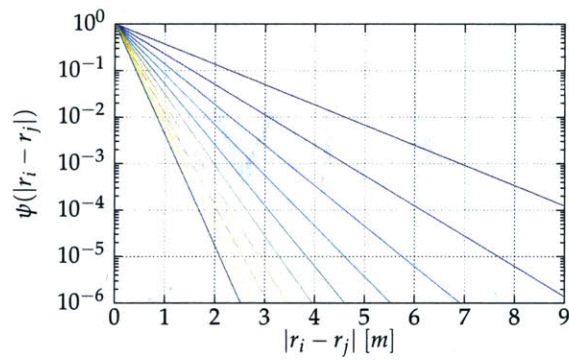
Figure 3-24: Exponential transition functions with λ ranging from $\frac{1}{2}$ to 5.

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
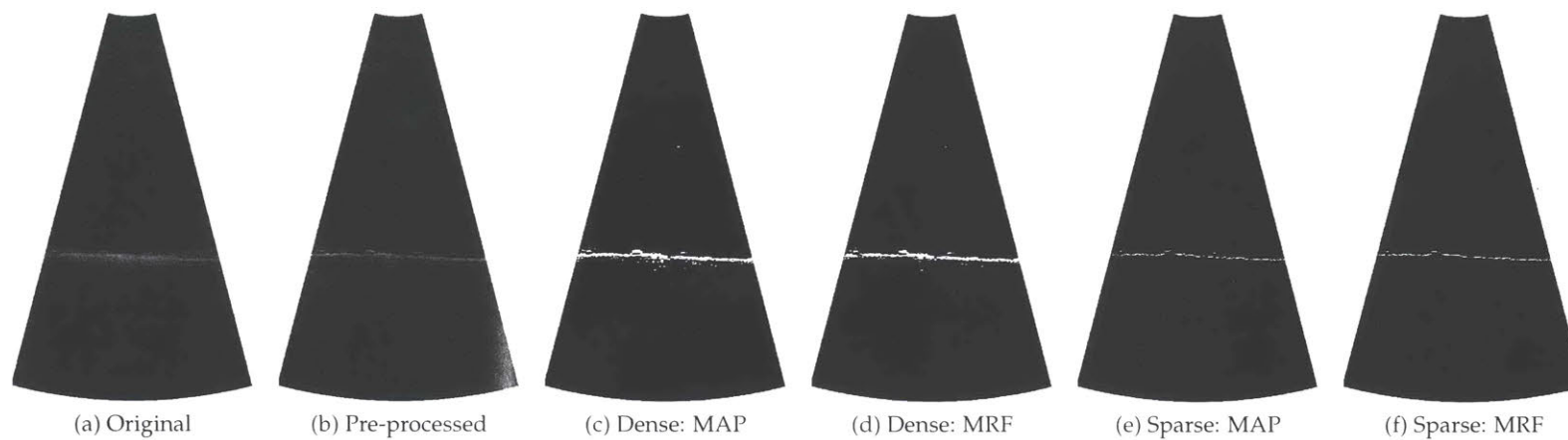
Figure 3-25: Segmentation results: [2018-03-14.00/36]

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
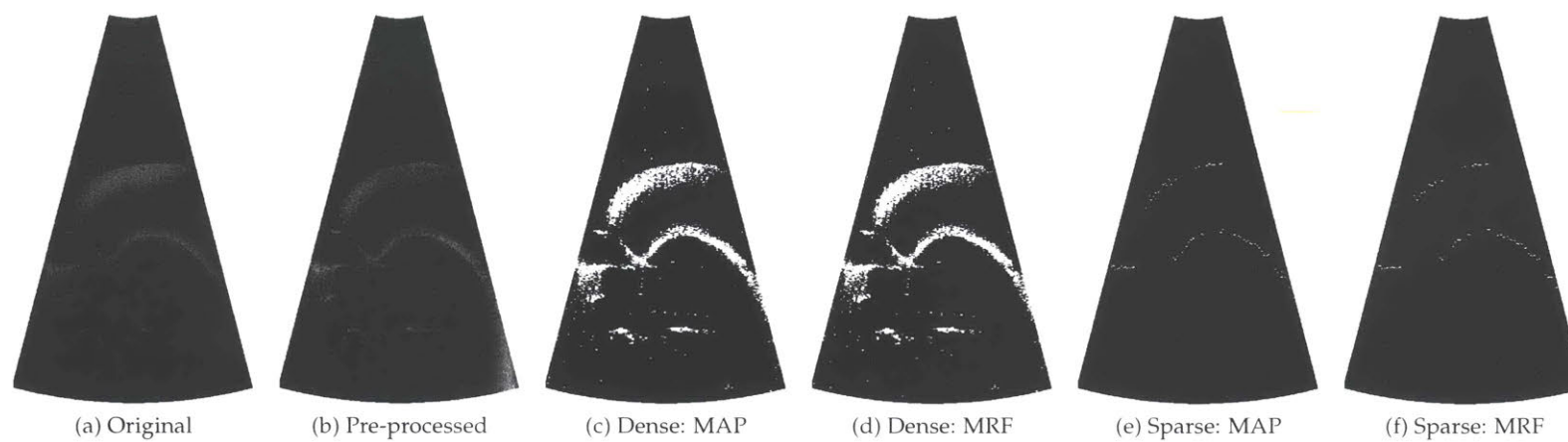
Figure 3-26: Segmentation results: [2018-03-14.00/305]

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
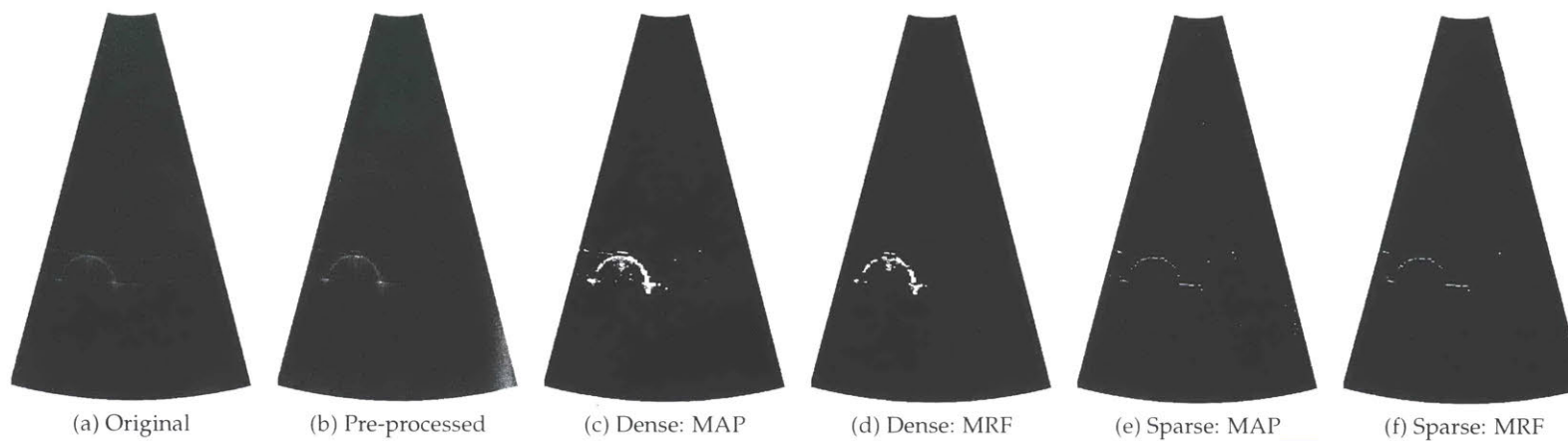
Figure 3-27: Segmentation results: [2018-03-14.00/338]

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
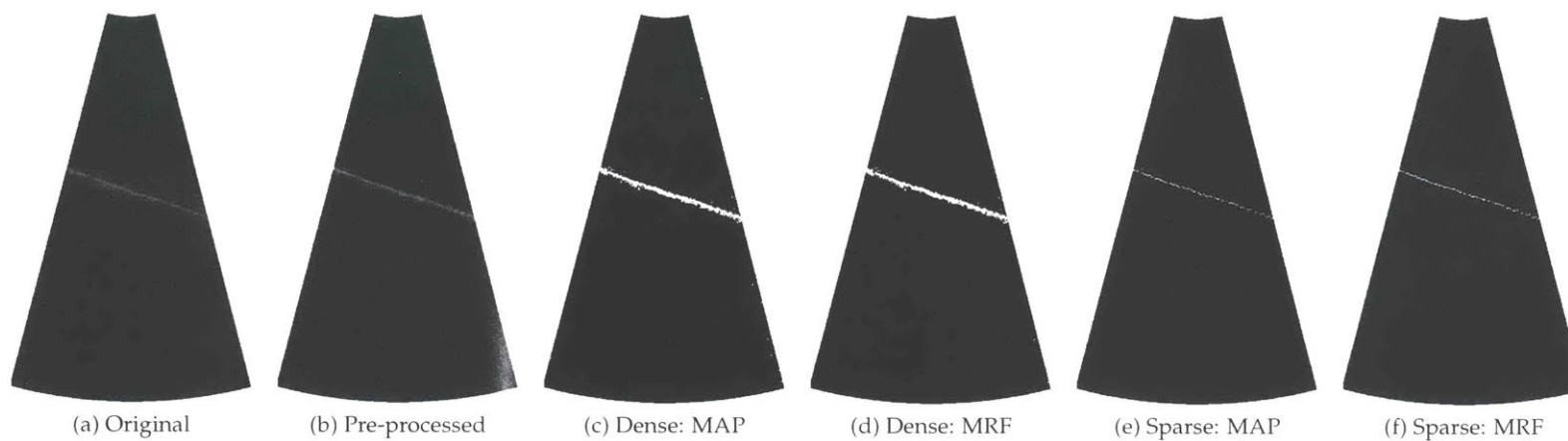
Figure 3-28: Segmentation results: [2018-03-14.00/1096]

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
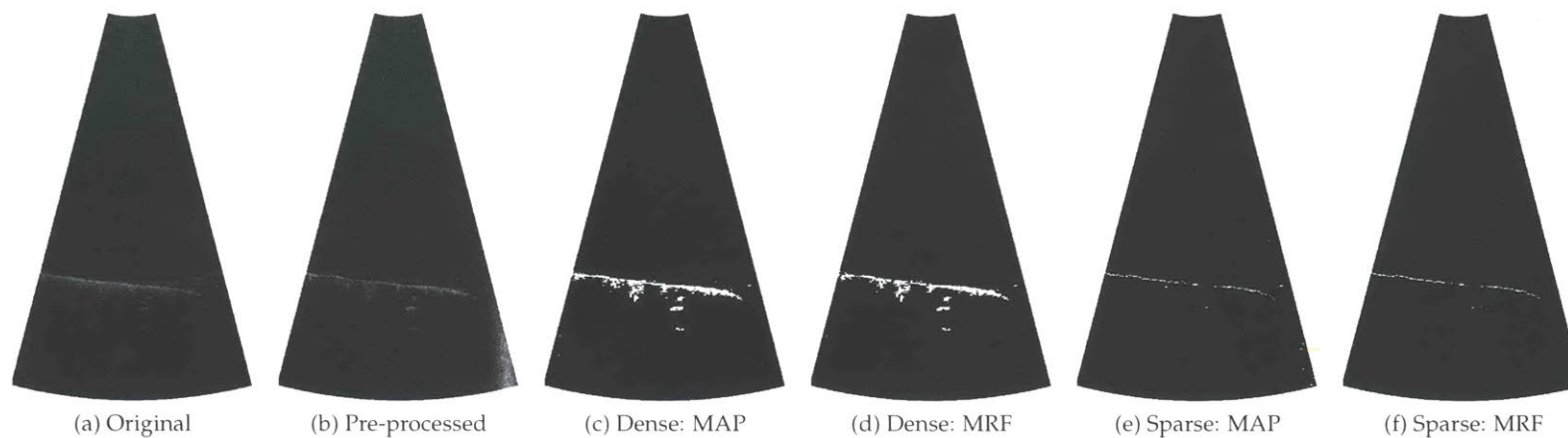
Figure 3-29: Segmentation results: [2018-03-14.00/1578]

| (a) Original | (b) Pre-processed | (c) Dense: MAP | (d) Dense: MRF | (e) Sparse: MAP | (f) Sparse: MRF |
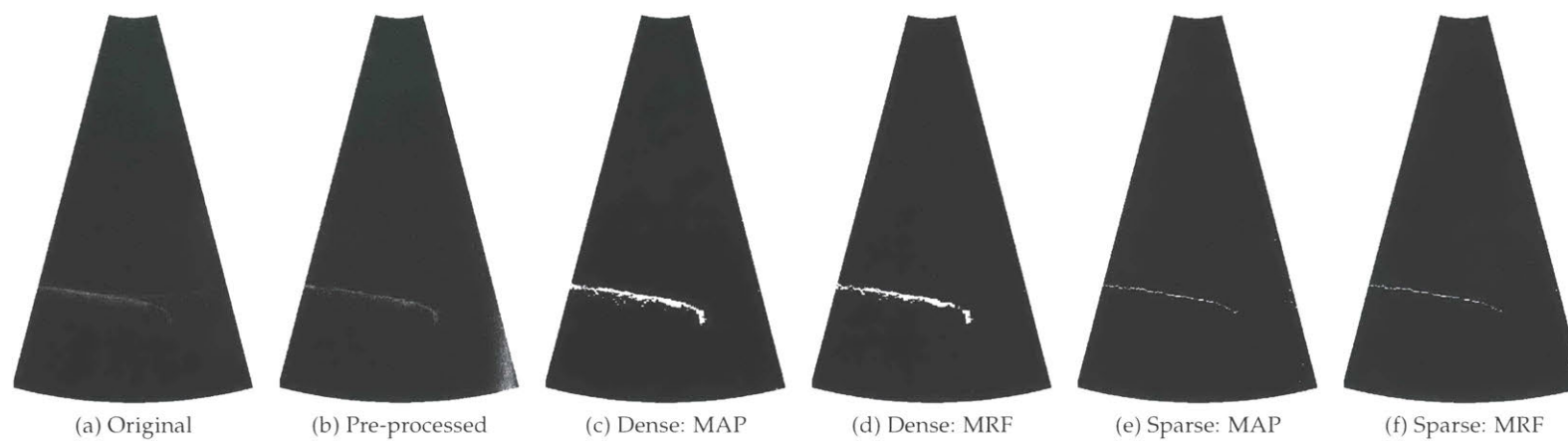
Figure 3-30: Segmentation results: [2018-03-14.00/2981]

3.8 Summary

We presented a set of techniques to pre-process and segment sonar data to derive range measurements to objects in the sonar field of view. Pre-processing techniques aim at mitigating the effect of noise and other artifacts present in the sonar data, and is informed by both the operating principles behind the sonar and simple acoustic models. To tackle the segmentation problem we began by proposing a simple but powerful mixture model to capture the background and object intensity distributions. The use of this model enabled us to improve upon the fixed-threshold segmentation approaches typically found in many mapping applications due to its increased robustness to variations in the scene and the resulting change in the received acoustic intensity. We then built upon these results by considering both simpler and more complex models: the sparse and dense MRF techniques, respectively.

Even though the results presented in this chapter focus on a particular multibeam sonar, they can easily be applied to other sonars, provided that they are adequately characterized, *i.e.*, that their impulse response, taper function, and mapping between beam number and angle are approximately known.

Chapter 4

Submap-based Simultaneous Localization and Mapping

4.1 Introduction

Multibeam profiling sonars, as shown in the previous chapter, have a narrow field of view, not unlike a line scanner. This limitation precludes the use of scan-matching techniques on scan pairs to obtain full pose constraint estimates; alignment between two scans, provided the two are in the same scanning plane, can only inform about the motion of the vehicle along that plane. This approach has a significant operational limitation: most (if not all) of the drift in the vehicle's position estimate will occur in the horizontal plane, so addressing this would require configuring the sonar to have overlapping horizontal scans for a significant part of its trajectory. This would severely limit the coverage rate of the inspection platform which, along with inspection accuracy and total coverage, is a key performance indicator for many inspection and mapping applications. Moreover, this approach would likely have limited success addressing long-term drift due to the lack of sufficient information in a sonar scan to localize the platform—the limited field of view combined with the repetitiveness of many underwater scenes (e.g. pilings, ship hulls) would make the data association problem significantly harder.

To address these issues, this chapter builds upon the results of the previous one to enable the use of scan-matching techniques to work fully in three-dimensional underwater scenes. This is accomplished through the use of *submapping* techniques, which take as input sonar range measurements and uncertain pose estimates and, over short time-spans, assemble three-dimensional maps. The temporal length of each submap is chosen such that it can be considered self-consistent. When certain criteria are fulfilled, these maps are then used to determine pairwise pose constraints, over both short and long time spans.

As an additional benefit, this approach has the potential of significantly reducing the scale of the mapping problem, as the number of variables is equal to the number of submaps, which can be orders of magnitude smaller than the number of individual sonar scans.

4.2 Related Work

Some of the earliest work on sonar-based underwater mapping employed both ship- and ROV-mounted multi-beam sonars to create bathymetric maps and inspect ship wrecks [101]. The proposed approach used occupancy grids to model not only the non-zero beam width of a sonar, but also the uncertainty associated with the vehicle pose.

More recently, vehicles equipped with scanning sonars have been used to inspect and map both man-made and unstructured environments. The mechanical scanning nature of these sonars, combined with their slow scanning speeds (on the order of a few degrees per second) poses a major operational challenge to the map-building effort: platforms will often hover in-place or even sit on the floor as a full 360° scan is assembled, which might not always be feasible [61, 80]. Once scans are assembled, scan-matching algorithms such as Iterative Closest Point (ICP) or probabilistic iterative correspondence (PIC) are used to obtain an estimate relative to either the previous scan (filtering) or an earlier scan (smoothing). These relative motion estimates are then incorporated in a pose estimation framework such as a particle filter [61] or an augmented-state Extended-Kalman Filter [60]. To avoid the operational constraints placed by the scanning nature of these sonars, some techniques rely on dead-reckoning estimates, provided by a kinematic model or a DVL-aided INS, to assemble the scans with while the vehicle is in motion.

While submapping techniques appear out of necessity in underwater applications, due to the previously described sensor limitations, they can also be found in terrestrial applications, where they are often motivated by the need to reduce the computational burden associated with large scale maps. One notable example is the *Atlas* framework, a submap-based architecture aimed at simultaneous localization and mapping applications in large, cyclic environments [9, 10]. In order to minimize the computational cost of SLAM, it uses submaps with bounded complexity—a limited number of poses and/or features (depending on the SLAM technique employed at the submap level). Similarly, in *Tectonic SAM*, a submap-based approach is used to avoid the computational cost associated with the linearization step [67]. The full factor graph is partitioned into smaller maps, which are optimized independently. Once this step is complete, and as in the *Atlas* framework the full map is obtained by optimization over the submap graph.

Scan matching techniques have also been experimentally demonstrated on micro-bathymetric mapping of natural, unstructured scenes using data from an ROV-mounted multibeam

sonar [85]. One of the key advantages of bathymetric applications is that the scene can be considered 2.5 dimensional; in other words, it can be described by a two-dimensional grid where each element stores the elevation value. This representation enables the use of image registration techniques such as 2D correlation in tandem with the previously mentioned ICP-type algorithms.

As a likely consequence of the prohibitively high cost of a multibeam sonar and DVL-aided INS, in combination with the computational cost of the associated mapping and estimation algorithms, fully three-dimensional underwater mapping techniques have only begun to appear recently in the literature. One notable example of such a platform is the *DEPTHX* vehicle, which was equipped with a total of 54 pencil-beam sonars (2° beam width), arranged in three rings oriented perpendicular to each of the three dimensions. This platform was used to inspect and map flooded sinkholes in Mexico. The SLAM method demonstrated in that application used particle filters, in combination with a highly specialized occupancy grid implementation to allow for fast execution of the numerous copy and update operations required by those filters [25, 26].

4.3 Problem Statement

To build a map of the scene, we must first estimate the positions **p** of the acoustic returns derived from the sonar scans in the global frame. Doing so requires estimating the pose of the sensor itself, **x**. The inputs to our estimation problem are the range estimates, derived from the multibeam sonar scans (chapter 3), and navigation data. As stated in section 1.6, the latter is assumed to be available in real-time from the platform's navigation payload in the form of full (6 DoF) odometry estimates $\mathbf{x}_i$.

4.4 Submap Assembly

4.4.1 Sonar Measurements

At every instant t_i, a new set of range and azimuth measurement pairs $(\tilde{r}_j, \tilde{\alpha}_j)$ is available, where $j \in 1, \ldots, N_i$ and N_i is the number of measurements received at that instant. Each measurement can be expressed as a point in the sonar coordinate frame

$$^{S}\tilde{\mathbf{p}}_j = \begin{bmatrix} \tilde{r}_j \cos \tilde{\alpha}_j \\ \tilde{r}_j \sin \tilde{\alpha}_j \\ 0 \end{bmatrix} \tag{4.1}$$

and transformed to the platform frame using the estimate of the transformation between sonar and platform:

$$^P\tilde{\mathbf{p}}_j = {}^P_S T\, {}^S\tilde{\mathbf{p}}_j \tag{4.2}$$

Having an accurate estimate of this transformation is critical, for the reasons detailed in chapter 2: a small error in the angular offset will result in a large registration error due to the long lever-arm effect created by the offset between the platform and sensor reference frames. As stated in section 1.6, for the remainder of this chapter we assume that one is available.

Using $^P\mathcal{P}_i$ to denote the set of all such points measured at time t_i, we write

$$^P\mathcal{P}_i = \bigcup_j {}^P\{\tilde{\mathbf{p}}_j\} \tag{4.3}$$

4.4.2   Odometry

Spatially registering the points requires knowledge of the platform pose in the global frame at that instant, $^W\hat{\mathbf{x}}_i$. While the sonar and navigation payload will output measurements at different time instants, it is often the case that the latter will produce pose estimates at a much higher rate than that of the sonar, so the time difference between the two should be minimal. In any event, an estimate for the platform pose at t_i should be available through interpolation of neighboring pose estimates. The globally registered points can then be obtained

$$
\begin{aligned}
^W\mathcal{P}_i &= {}^W_P T_i\, {}^P\mathcal{P}_i \\
&= \bigcup_j \left({}^W_P T_i\, {}^P\tilde{\mathbf{p}}_j \right)
\end{aligned}
\tag{4.4}
$$

The odometry-based pose estimate at t_i can also be described as the composition of a series of relative odometry estimates, $^i_{i-1}T$:

$$^j_i T = \prod_{k=i}^{j-1} {}^{k+1}_k T \tag{4.5}$$

Similarly, given a reference pose $^W\hat{\mathbf{x}}^P_R$, the platform pose at time t_i, can be recovered from the composition of the relative odometry estimates:

$$^W\hat{\mathbf{x}}^P_i = \left(\prod_{k=R}^{i-1} {}^{k+1}_k T \right) {}^W\hat{\mathbf{x}}^P_R \tag{4.6}$$

This can be used to obtain all the points in that segment, referenced to the same initial pose:

$$\mathcal{M}_j = \bigcup_i \left({}^j_i T \mathcal{P}_i \right) \tag{4.7}$$

where i indexes all the poses in the segment. The expression above defines a *submap* as a collection of points expressed with respect to the same reference (base) pose: the platform pose at time t_j. Note that while we have chosen the first pose in the segment as the reference, the formulation above is equally valid if any other pose in the segment is chosen as the base pose.

4.4.3 Self-consistency

As described in chapter 2, as the length of a trajectory segment increases, so does the uncertainty in the pose estimate. When referenced to the base pose, the uncertainty in the poses along that segment will grow from Σ_0, the uncertainty in the estimate of $\mathbf{x}_0$ (the reference pose) in that segment. Since we are only concerned with the uncertainty of a given point with respect to the reference pose, we can set $\Sigma_0 = 0$ and compute Σ_i according to equation 2.16. The uncertainty associated with the ith pose estimate in segment, $\mathbf{x}_i$, is

$$\Sigma_{\mathbf{x}} \approx \sqrt{t_i - t_0}\, \Sigma \tag{4.8}$$

Considering a range measurement taken from pose i, the uncertainty in the position of that measurement with respect to the base pose can be written as

$$\Sigma_{\mathbf{p}} = \Sigma_{\mathbf{x}} + J_A{}^P \Sigma_p J_A^T \tag{4.9}$$

To keep a submap self-consistent, we must then ensure that the uncertainty of each of the points within it is below a pre-specified upper bound, Σ_{max}, which, as previously mentioned, will be specified by the end-use of the scene model that is to be produced. Replacing $\Sigma_{\mathbf{p}}$ in (4.9) with this threshold we can solve for the maximum allowable length of a submap.

4.5 Submap Representation

In the description above we have considered a point-based representation of the map, as points provide a useful parameterization of the sonar range measurements when moving between reference frames and keeping track of the associated uncertainty. Still, there are other representations which can prove more useful for the purposes of constructing an

accurate scene model from sonar range measurements.

4.5.1 Occupancy Grids

One of the limitations of point-based representations is that they do not capture the "negative information" present in a range measurement—that is, it does not capture the fact that the space between the sensor and the scattering object is free of other objects. At the same time, these representations do not explicitly model uncertainty regarding the actual presence of an actual object. Instead, only the uncertainty associated with its position in space is modeled, as we have seen from eq. (4.9); the existence of the scattering object is considered *certain*.

This limitation becomes significant when creating a map from range measurements which are likely to have outliers—point $\mathbf{p}$ may, after all, result from an artifact in the sonar scan, or from a dynamic object, such as a fish or a bubble, crossing its field of view. Moreover, while a point-based representation can account for uncertainty in the azimuth and elevation directions (subsection 2.2.1), there is an implicit assumption that these uncertainties are small.

Occupancy grid techniques provide a way to model both the negative information and the angular uncertainty associated with a range measurement—this is accomplished by discretizing the scene into a volumetric grid and tracking, for each element (*voxel*), its probability of occupancy [21]. Letting o denote the occupancy of a grid element ($o(x_i, y_i, z_i) = 0$ if the grid element at x_i, y_i, z_i is empty and $o(x_i, y_i, z_i) = 1$ if it is occupied), the occupancy probability is updated according to Bayes' rule [108]

$$p(o|z_{0:i}) = \frac{p(z_i|z_{0:i-1}, o)}{p(z_i|z_{0:i-1})} p(o|z_{0:i-1}) \tag{4.10}$$

where $z_{0:i}$ denotes all range measurements z from $t = t_0$ to $t = t_i$. Assuming a static map and conditional independence between grid elements, the occupancy probability update can be expressed as (after simplification)

$$\frac{p(o|z_{0:i})}{1 - p(o|z_{0:i})} = \frac{p(o|z_i)}{1 - p(o|z_i)} \tag{4.11}$$

Using $l(e)$ to denote the logarithm of the ratio $\frac{p(e)}{1-p(e)}$ (also known as the *log-odds* of event e), and assuming an initial prior probability of $p(o) = 1 - p(o) = 0.5$, this can be further simplified to

$$l(o|z_{0:i} = l(o|z_{0:i-1}) + l(o|z_i) \tag{4.12}$$

The term $p(o|z_i)$ is the (inverse) sensor model: it describes the probability of occupancy of

an element given a measurement, and it takes different values depending on how the element being updated relates to the measurement. For a sensor positioned at $\mathbf{x}$ and a range measurement corresponding to a point $\mathbf{p}$ (eq. 2.1), the voxels v along the line connecting the two are updated according to

$$p(o|z_i) = p_{\text{miss}} \quad (\text{for } v \in \mathcal{V}_{\text{empty}}) \tag{4.13}$$

This is the probability of that voxel being occupied given that the measured range was greater than the distance to the sensor (in other words, the sensor *missed* the object in that voxel). Similarly, for voxels at the endpoint of the range measurement ($v \in \mathcal{V}r$)

$$p(o_v|z_i) = p_{\text{hit}} \quad (\text{for } v \in \mathcal{V}_r) \tag{4.14}$$

The sets of voxels whose occupancy probabilities must be updated, $\mathcal{V}_{\text{free}}$ and $\mathcal{V}_r$, are determined through ray tracing from the sensor origin. It is during this step that the non-zero field of view of a sonar must be taken into account, requiring the computation of multiple rays spanning the constant-range spherical rectangle ("range cap") where the scattering object(s) may lie. Care must be taken to only update voxels once both sets have been computed, as voxels close to the sensor origin will be crossed by many rays [36].

4.5.2 Point Clouds

Once the occupancy grid has been updated with all the range measurements in that segment, it can be converted to a point cloud by taking the centroid of each voxel for which the probability of occupancy is greater than a threshold. Typically, $p > 0.5$ is used, but higher values can be used if higher confidence in the resulting point cloud and/or increased robustness to outliers is required.

Submap pre-processing

Despite the use of robust classifiers and volumetric techniques, it is reasonable to assume that there will still be outliers in the point cloud representation of the submap. To reduce the number of outliers in a submap, so as to improve the accuracy of the model and mitigate their effect on the registration process, the submap must be filtered before it is added to the map. It is important to note that any filtering process must take into account the features of interest in the scene; certain filtering processes may be very effective at removing outliers, but do so at the expense of small-scale detail in the scene [115].

Clustering Filter   Other than noise and artifacts in the sonar scan, range measurement outliers can also be caused by valid returns from actual scatterers such as bubbles, suspended sediments, and fish. For scenes such as harbors or berthed ships, these are particularly common, highlighting the need for filtering. A cluster-based filter works by grouping points into clusters based on their distance to a nearest neighbor: starting from a set $\mathcal{P} = \{\mathbf{p}\}$, all points within a distance ϵ of those in $\mathcal{P}$ are added to the set, and the process is repeated until no neighboring points are found. Once clusters are identified, those below a certain number of points (or other relevant criteria, such as planarity) can be removed.

Normal Estimation   Once outlier clusters have been removed, the next step is to estimate the normals associated with each point. We choose to perform this step at this point in the pipeline (as opposed to when deriving range measurements) due to the narrow field of view of the sonar: from a single sonar scan, we can estimate at most a two-dimensional projection of the surface normal.

4.6   Pairwise Registration

4.6.1   Candidate Selection

To identify pairwise registration candidates, the simplest approach is to attempt to pairwise register $\mathcal{M}_i$ with all $\mathcal{M}_j$ such that $j < i$. This can quickly become computationally expensive as the number of submaps in a session grows. To avoid this we use a technique analogous to a *chart*: once the current session spans a minimum number of submaps, we use the full map to determine the scene's principal plane[1]. Each submap $\mathcal{M}_i$ is then projected onto this plane to determine which of its cells it spans—the principal plane is discretized into square cells with side length δ_C, and a submap $\mathcal{M}_i$ is said to cover (or span) a cell if that cell contains a minimum number of points in $\mathcal{M}_i$. This chart can be updated as new submaps are added, or rebuilt from scratch if needed (e.g. if the current principal plane is no longer a good fit for the scene). Once a new submap is available, it is added to the chart, which is then queried to identify candidates for registration.

4.6.2   Point Cloud Registration

After the candidates have been identified, the submaps can be aligned using the *Iterative Closest Point* (ICP) algorithm [7]. This algorithm first determines the pairwise correspondences between pairs of points in the *target* and *source* sets. Once these have been com-

---

[1]This step is made easier if a prior scene model is available and has successfully been registered with the current world frame, as the principal plane of the scene is now known in advance.

puted, it then estimates the rigid body transform that best aligns the set of pairs, where the alignment is captured in an objective function $h(\cdot)$. The source point cloud is then transformed using the latest estimated and the process is repeated, usually until the stopping criteria are fulfilled. Common stopping criteria include: (*i*) a maximum number of iterations, (*ii*) the change in the transformation estimate (with respect to the previous estimate) is below a threshold, (*iii*) the objective function is below a threshold value (e.g. the average distance between corresponding pairs of points is less than ϵ).

Several variations on the Iterative Closest Point algorithm have been proposed since it was first presented [90]. In the standard version of ICP, the objective function $h(\cdot)$ minimizes the pairwise *point-to-point* distance between corresponding points:

$$\hat{T} = \arg\min_{T} \sum_{(i,j)\in\mathcal{C}} \|\mathbf{p}_i - T\mathbf{p}_j\|^2 \tag{4.15}$$

For point sets where point normals $\hat{\mathbf{n}}_i$ are available, this formulation has been improved upon by the *point-to-plane* version of the same algorithm. In it, the objective function is modified to depend solely on the point-to-plane distance between point pairs, *i.e.*, the projection of the point-to-point difference vector, $\mathbf{p}_i - T\mathbf{p}_j$ along the point normal $\hat{\mathbf{n}}$

$$\hat{T} = \arg\min_{T} \sum_{(i,j)\in\mathcal{C}} |(\mathbf{p}_i - T\mathbf{p}_j)\cdot\hat{\mathbf{n}}|^2 \tag{4.16}$$

This modification reflects the assumption that the point sets are, in fact, noisy samples of some surface. In *generalized* ICP [95], a similar assumption is made: the two point sets are treated as realizations of two unknown, normally-distributed point sets with covariance matrices Σ_i^T and Σ_i^S. In this case, the objective function is in fact the MLE of the transformation T, and written as the sum of *Mahalanobis* distance between two corresponding points $\mathbf{q}$ and $T\mathbf{p}$,

$$\hat{T} = \arg\min_{T} \sum_{(i,j)\in\mathcal{C}} (T\mathbf{p} - \mathbf{q})^T \left(\Sigma + T\Sigma T^T\right)^{-1} (T\mathbf{p} - \mathbf{q}) \tag{4.17}$$

As the name implies, equation 4.17 can be shown to be the generalization of the objective functions of both the point-to-point (eq. 4.15) and point-to-plane (eq. 4.16) ICP variants [95]. The general form of the ICP technique is summarized in algorithm 1.

4.7 Pose Graph Formulation

Having described both the submap assembly and pre-processing techniques, as well as the methods used to derive pairwise constraints between submaps, we now turn to the task of

---

Algorithm 1: Iterative Closest Point algorithm [7, 95]

Input: Source point set $\mathcal{P}_S$, target point set $\mathcal{P}_T$, initial transformation estimate T_0, maximum correspondence distance ϵ

Output: Rigid body transformation estimate, T, correspondence set $\mathcal{C}$

1 **begin**
2      $T \leftarrow T_0$
3   **repeat**
4          $\mathcal{C} \leftarrow \varnothing$
5   **foreach** $\mathbf{q_j} \in \mathcal{P}_S$ **do**   ▷ Determine correspondences
6
7              $\mathbf{p}_i \leftarrow \arg\min_{\mathbf{p} \in \mathcal{P}_T} \|T\mathbf{p} - \mathbf{q}\|$ ▷ Find closest point in $\mathcal{P}_T$
8   **if** $\|T\mathbf{p}_i - \mathbf{q}_j\| \leq \epsilon$ **then**
9                 $w_i \leftarrow 1$
10                $\mathcal{C} \leftarrow \mathcal{C} \cup (i, j)$ ▷ Add correspondence to $\mathcal{C}$
11   **else**
12                 $w_i \leftarrow 0$
13   **end**
14   **end**
15          $T \leftarrow \arg\min h()$ ▷ Estimate rigid-body transformation
16          $\mathcal{P} \leftarrow T\mathcal{P}$ ▷ Transform point cloud
17   **until**
18   **return** $T, \mathcal{C}$
19 **end**

---

formulating it as a pose estimation problem.

4.7.1   Odometry

Given two subsequent odometry-based pose estimates, $\hat{\mathbf{x}}_{i-1}$ and $\hat{\mathbf{x}}_i$, we can obtain a relative odometry measurement expressing the latest pose with respect to the previous pose, $^{i-1}\mathbf{x}_i$

$$
\begin{aligned}
\tilde{\mathbf{z}}_{i-1,i} &= {}^{w}\tilde{\mathbf{x}}_i \ominus {}^{w}\tilde{\mathbf{x}}_{i-1} \\
&= {}^{i-1}\mathbf{x}_i + \mathbf{v}_i^o \\
&= [\delta_x\ \delta_y\ \delta_z\ \delta_\psi\ \delta_\theta\ \delta_\phi]^T + \mathbf{v}_i^o
\end{aligned}
\tag{4.18}
$$

Since only the x, y and ψ components of the estimates $\hat{\mathbf{x}}_{i-1}$ and $\hat{\mathbf{x}}_i$ are relative, we opt instead to express the odometry-based estimates as a combination of three measurements describing the depth and attitude of the platform, as well as the relative planar odometry (with respect to the previous pose). This way, the trajectory can be represented as a pose graph, where a sequence of poses is connected by relative planar odometry measurements, $\tilde{\mathbf{z}}_{i-1,i}^o$, as well as absolute attitude and depth measurements $\tilde{\mathbf{z}}_i^a$ and $\tilde{\mathbf{z}}_i^z$, as illustrated in figure 4-1.

Planar Odometry

The position components of the estimates in eq. 4.18, $\hat{\mathbf{x}}_i$, describe the position x, y, z of the local-level frame at time t_i —out of these, only x and y are obtained through integration of platform velocity and, as such, accumulate uncertainty over time. Similarly, of the orientation components in $\mathbf{x}$, describing the orientation of the platform at time t_i with respect to the local-level frame at that time, only the yaw component ψ will grow more uncertain over time[2]. The measurement model for this relative planar odometry is

$$
\begin{aligned}
\tilde{\mathbf{z}}_{i-1,i}^o &= h\left(\mathbf{x}_{i-1}, \mathbf{x}_i\right) + \mathbf{v}_i^o \\
&= \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_\psi \end{bmatrix} + \mathbf{v}_i^o
\end{aligned}
\tag{4.19}
$$

$\mathbf{v}_i^o$ is the zero-mean, normally distributed measurement noise with covariance Σ_o which, provided that platform motion is uniform between two updates, will remain approximately constant between two updates. Note that this corresponds to the navigation model

---

[2]This statement will not hold for systems where the navigation payload relies on a *gyro-compass* to obtain its orientation estimate, as the accuracy of the yaw estimate will not degrade over time. In such cases, the heading estimate should also be considered an absolute measurement and lumped with the attitude measurement (eq. 4.20).
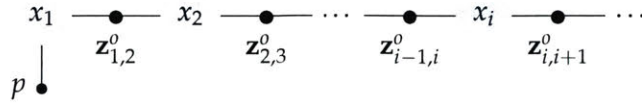
$$x_1 \;—\!\!\bullet\!\!—\; x_2 \;—\!\!\bullet\!\!—\; \cdots \;—\!\!\bullet\!\!—\; x_i \;—\!\!\bullet\!\!—\; \cdots$$

$$\mathbf{z}^o_{1,2} \qquad \mathbf{z}^o_{2,3} \qquad \mathbf{z}^o_{i-1,i} \qquad \mathbf{z}^o_{i,i+1}$$

$$p \;\bullet$$

Figure 4-1: Odometry chain: relative planar odometry measurements $\mathbf{z}^o_{i-1,i} = [\delta_x \; \delta_y \; \delta_\psi]^T$, describing the displacement in the local-level frame, connect sequential poses $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$. Depth and attitude measurements, z^d_i and $\mathbf{z}^a_i = [\theta_i \; \phi_i]^T$, model the absolute measurements at $\mathbf{x}_i$. A prior p provides the initial estimate for $\mathbf{x}_0$.

described in subsection 2.2.3, so the diagonal terms of Σ_o will be mostly determined by the navigation payload method and parameters (typically the velocity and heading measurement performance on a DVL-aided INS).

Attitude

As previously mentioned, the pitch and roll components in the estimate $\hat{\mathbf{x}}_i$ can be considered absolute measurements, particularly for systems with slow dynamics such as inspection and mapping platforms, where we do not expect pitch and roll estimates to be biased due to the platform undergoing significant accelerations. The measurement model for the platform attitude is

$$\begin{aligned} \mathbf{z}^a_i &= h(\mathbf{x}_i) + \mathbf{v}^a_i \\ &= \begin{bmatrix} \theta_i \\ \phi_i \end{bmatrix} + \mathbf{v}^a_i \end{aligned} \tag{4.20}$$

The covariance for the zero-mean, normally-distributed measurement noise, is $\Sigma_a = diag([\sigma^2_\theta \; \sigma^2_\phi])$. Values for the standard deviations σ_θ and σ_ϕ range from $1°$ for typical off-the-shelf AHRS solutions [1] to $0.01°$ for a high-end INS [78].

Depth

Depth estimates can be obtained from pressure measurements through the use of the seawater models, as described in the previous chapter [29]. These provide depth as a function of salinity, temperature, and pressure.

$$\begin{aligned} z^d_i &= h(\mathbf{x}_i) + v^d_i \\ &= z_i + v^d_i \end{aligned} \tag{4.21}$$

Pressure measurement accuracy ranges from 0.5% to 0.01% of *full-scale* pressure which, for shallow water ($\leq 100m$) applications corresponds to 0.5 and 0.01m, respectively [4, 35].

4.7.2 Loop Closures

Having described the measurement models for the odometry-based pose estimates, we must now model the pairwise registration constraints from section 4.6. Given two submaps, $\mathcal{M}_i = \{\mathbf{q_i}\}$ and $\mathcal{M}_j = \{\mathbf{p_j}\}$, we can rewrite eq. 4.15 (or, equivalently, any of the other objective functions) as

$$\hat{T} = \arg\min_{T} \sum_{k} \left(w_k \|^w\mathbf{p}_k - T^w\mathbf{q}_k\|^2 \right) \tag{4.22}$$

where $w_k = 1$ if $\mathbf{p}_k$ corresponds to $\mathbf{q}_k$, and $w_k = 0$ otherwise. In the ideal scenario (perfect position estimate), $T \approx I$, as the two world-referenced submaps would be aligned. If we instead express both $\mathbf{p}$ and $\mathbf{q}$ with respect to their reference poses

$$\hat{T} = \arg\min_{T} \sum_{k} \left(w_k \|^i\mathbf{p}_k - T^j\mathbf{q}_k\|^2 \right) \tag{4.23}$$

then the optimal transformation estimate should be an estimate of $_j^iT$: $\mathbf{x}_j$ expressed with respect to $\mathbf{x}_i$. This is the same as $\mathbf{x}_j \ominus \mathbf{x}_i$, and depending on the technique used to determine $\hat{T}$, can correspond to either a full 6 DoF constraint like that of eq. 4.18, or to a partial constraint similar to eq. 4.19. For the same reasons mentioned in section 4.7.1, we opt for the latter, and model the pairwise registration as

$$\mathbf{z}_{i,j}^r = \mathbf{h}(\mathbf{x}_i, \mathbf{x}_j) + \mathbf{v}^r$$
$$= \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_\psi \end{bmatrix} + \mathbf{v}^r \tag{4.24}$$

4.7.3 Solution Techniques

The pose graph described above is formulated as an optimization problem, where we want to find the *maximum a posteriori* (MAP) estimate for the set of reference poses $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$

$$X_{MAP} = \arg\min_{X} \left(\sum_{i=1}^{N} \|f(x_{i-1}, u_{i-1}) - x_i\|^2 + \sum_{(j,k)} \|h(x_j, x_k) - z_{jk}\|^2 \right) \tag{4.25}$$

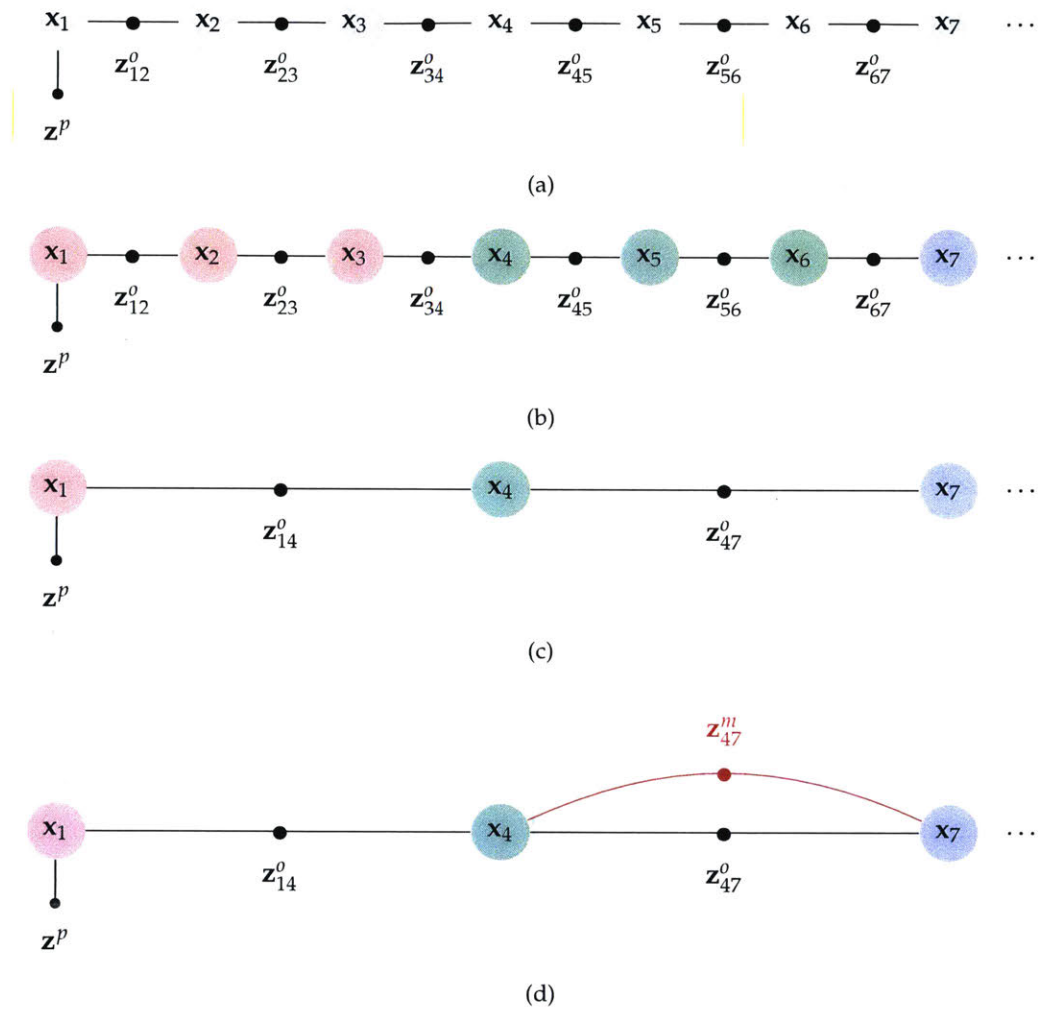where $f(\cdot)$ and $h(\cdot)$ describe the system dynamics and measurement models, respectively [18, 44].

Figure 4-2: Submap-based SLAM - pose graph construction. Starting from an odometry chain (4-2a), we group range measurements into submaps (4-2b); all the measurements in a submap are referenced to a base pose, and the odometry estimates within that segment are composed into a single measurement connecting that base pose to the next (4-2c). As new submaps become available, they are used to derive loop closures through pairwise registration with prior submaps (4-2d).

4.8 Experimental Results

In this section we present experimental results for the techniques described in sections 4.5 through 4.7

4.8.1 Platform

Navigation Payload

The data used in this section were collected using the HULS3 model of Bluefin Robotics *Hovering Autonomous Underwater Vehicle (HAUV)* [114, 37]. This particular model is equipped with five rim-driven thrusters configured to make it directly controllable in all axes except roll. Its navigation payload comprises an AHRS based on a Honeywell HG1700 inertial measurement unit [34], a 1.2MHz Teledyne/RDI Workhorse Navigator Doppler velocity log [124], and a Keller absolute pressure sensor [35]. While pose estimates are available from the aided INS, it does not support external updates or resets. As this platform was designed for hull inspection, the DVL sits in a basket that can rotate along an axis parallel to the vehicle's pitch axis, allowing for both hull- and bottom-relative tracking. All the results shown in this section use the DVL in bottom-tracking mode.

Mapping Payload

The HAUV's primary inspection and mapping payload is the dual-frequency identification sonar previously described in section 3.7. The sonar sits on an rotating mount inside the DVL basket, which can rotate 90°. To maximize the system coverage rate and detail, we fix the sonar forward[3]. (pointing along the body's x axis), and operate it exclusively in its higher frequency mode (1.8 MHz), with the provided "concentrator" lens.

4.8.2 Data sets

Hull Inspection: *USNS Mercy* (T-AH-19)

This dataset covers the inspection of the running gear of the *USNS Mercy* hospital ship, shown in figure 4-3. In it, the vehicle moves in constant-depth transects, roughly parallel to the ship's longitudinal axis inspecting the running gear from both port and starboard. The data was acquired over the course of approximately one hour and twenty minutes, spanning a total of 182 submaps, illustrated in figure 4-4.

---

[3]With the payload basket in the "half-split" configuration

(a) *USNS Mercy* hospital ship

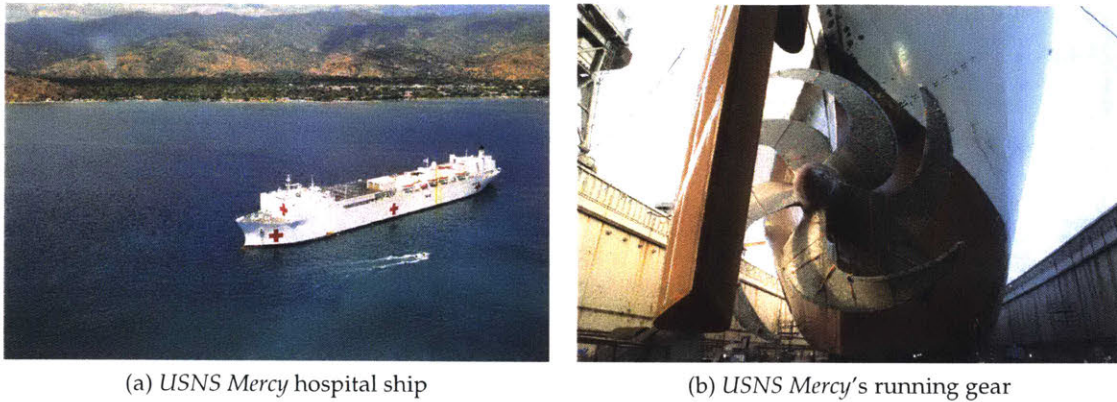(b) *USNS Mercy*'s running gear

Figure 4-3: The *USNS Mercy* hospital ship (T-AH-19) and its running gear, viewed from aft while in dry dock [sources: (a) US Navy, public domain; (b) Dave R., CC BY-NC 2.0].
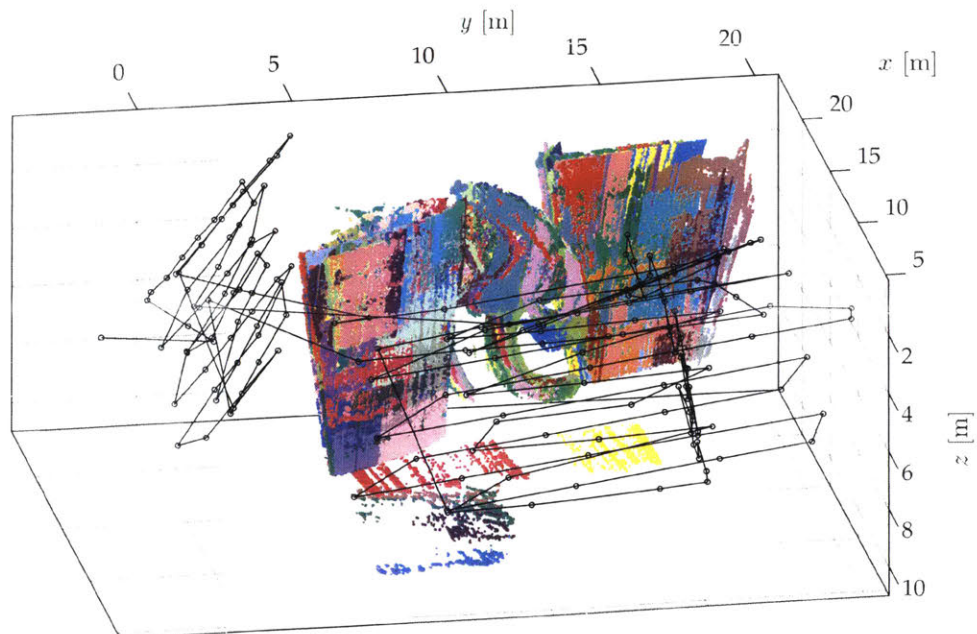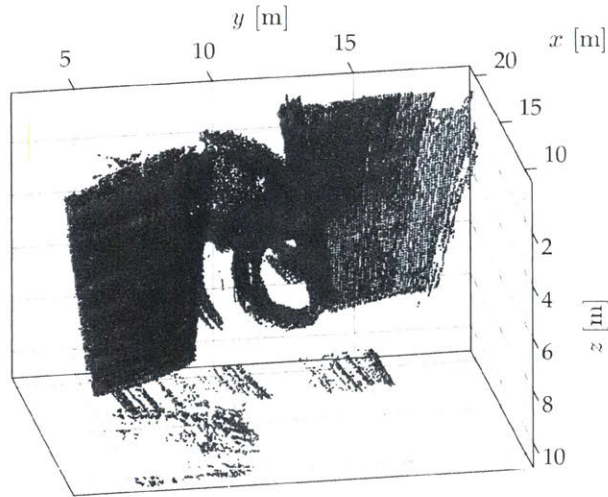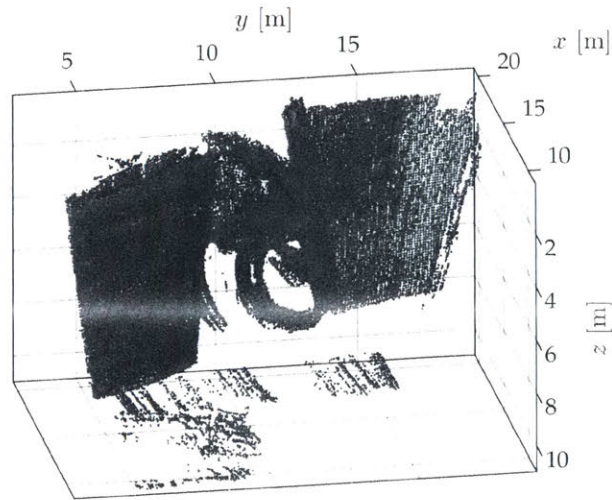

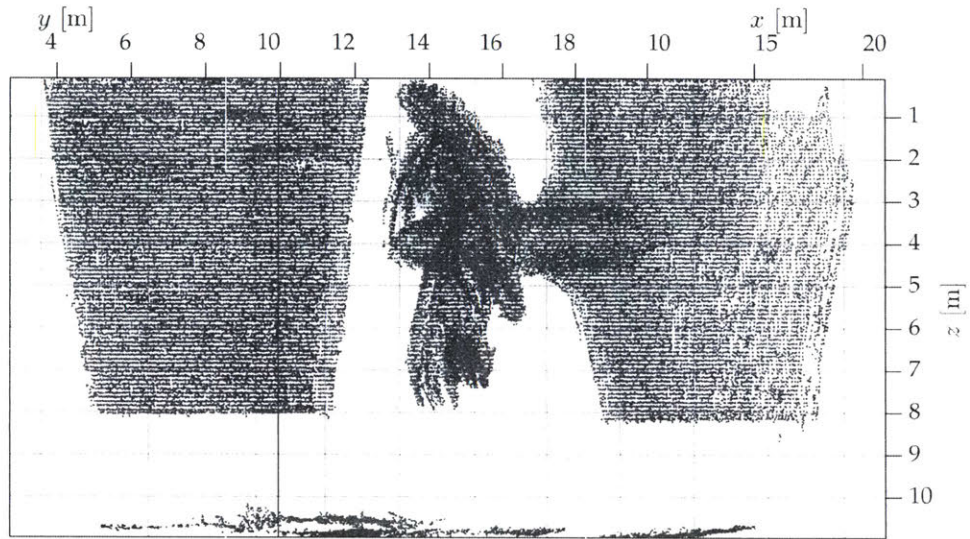Figure 4-4: *USNS Mercy*: vehicle trajectory and submap coverage.
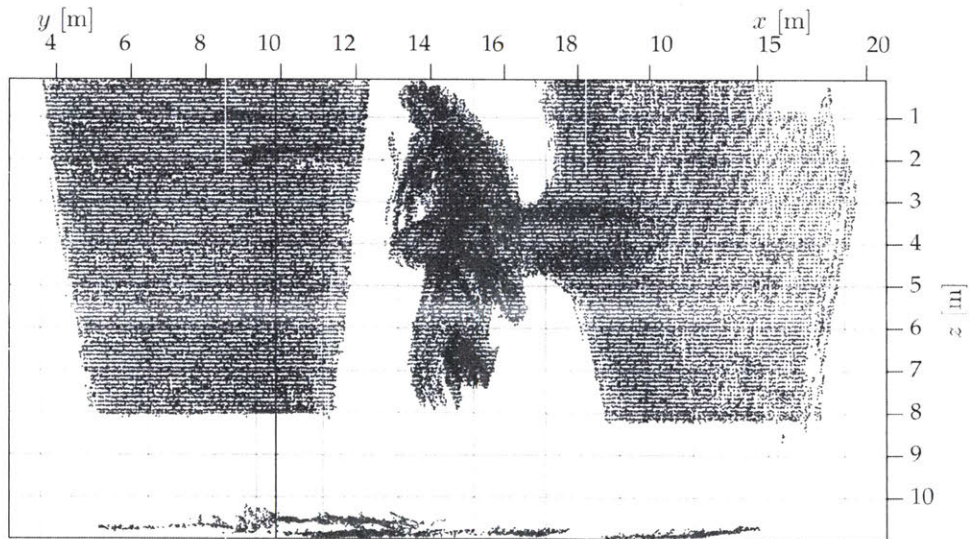
(a) Odometry

(b) SLAM

Figure 4-5: *USNS Mercy*: odometry- and SLAM-based maps (perspective view)

(a) Odometry

(b) SLAM

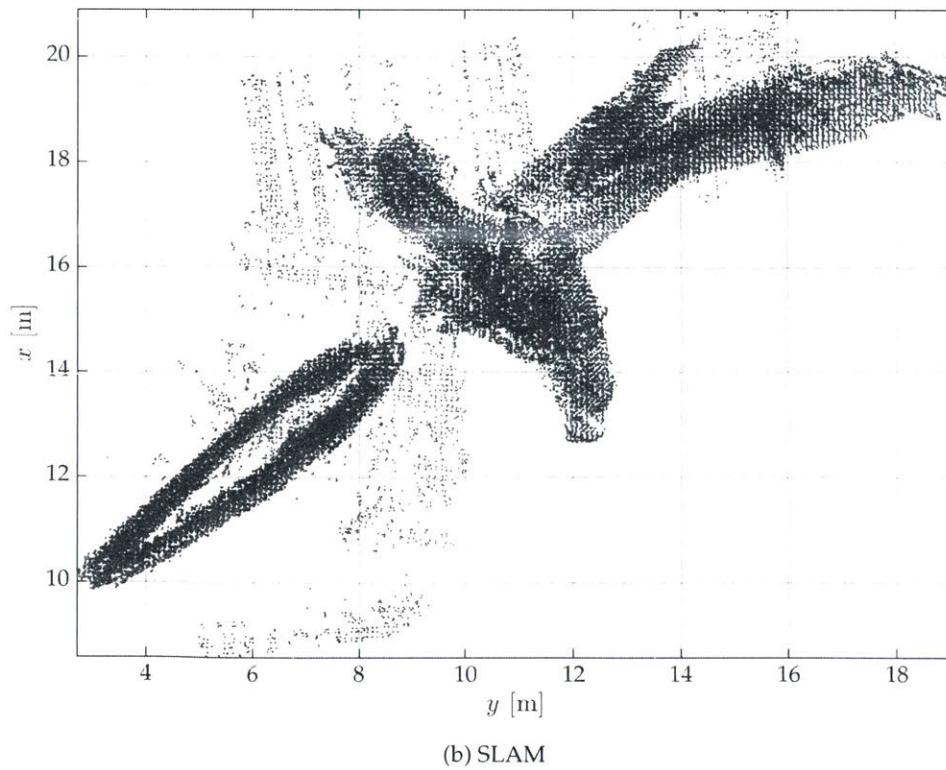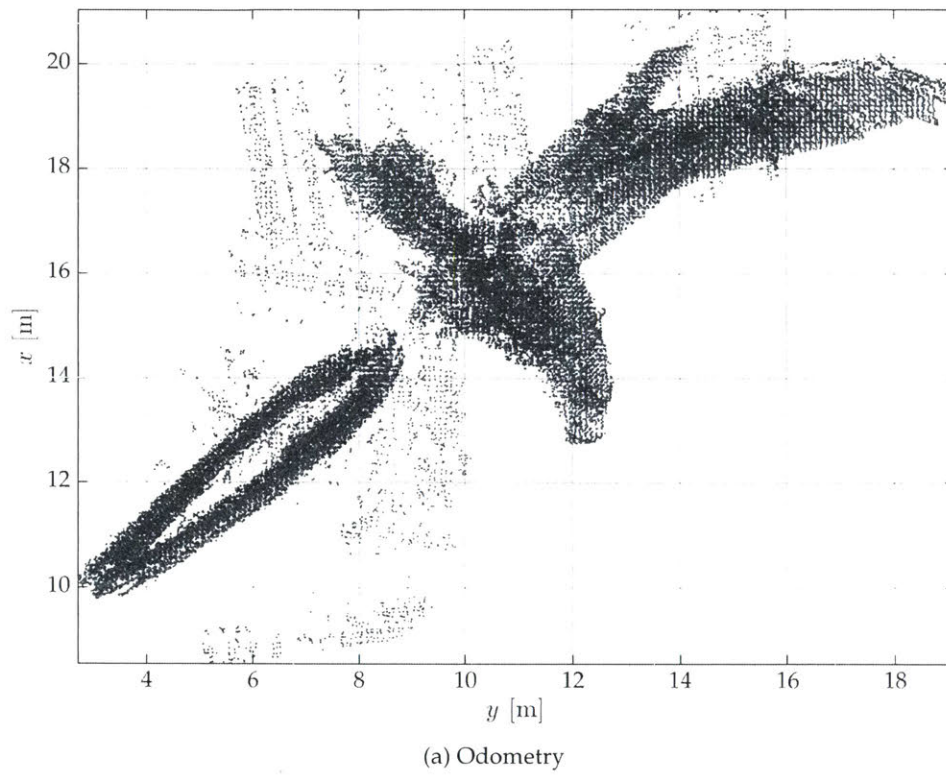Figure 4-6: *USNS Mercy*: odometry and SLAM-based maps (profile view)

(a) Odometry

(b) SLAM

Figure 4-7: *USNS Mercy*: odometry and SLAM-based maps (plan view)

Pier Inspection: Seawall, US Naval Base San Diego

This dataset covers the northeast corner of a seawall next to Pier 9 at the US Naval Base San Diego, where the *SS Curtiss* was berthed at the time. In it, the platform moves parallel to the seawall at about $0.2m.s^{-1}$ (sway), along an "L"-shaped trajectory, with an average stand-off distance of about 4 meters. It comprises a total of 60 submaps, each spanning approximately 20 seconds and a total of 150 platform poses. The results for this data set are shown in figures 4-8 through 4-11. Figure 4-8 shows a photograph of the seawall next to a sample submap from that dataset, where the corrugations are clearly visible. Figure 4-9 shows the coverage of the inspection trajectory: while there is significant repeat coverage throughout the trajectory, there is a noticeable gap at the corner. This was the result of the fast yaw adjustment maneuvers that were executed when moving between perpendicular segments in the inspection trajectory. Figures 4-10 and 4-11 show the odometry- and SLAM-based final maps, in both plan and perspective views. From the plan view of the map the effects of drift in the odometry estimate are visible: on the right-hand side of figure 4-10b there is a noticeable misalignment between the different passes; this has been corrected in the SLAM-based map in figure 4-11b.

4.9 Summary

This chapter described a submap-based approach to underwater SLAM. Submaps are assembled by accumulating sonar scans to capture sufficient information to enable the use of scan-matching techniques. At the same time, this accumulation happens over short time scales, to keep the submaps self-consistent.

The proposed method was experimentally demonstrated in real-time ship hull inspection applications, successfully mitigating drift in the platform's pose estimate and improving the accuracy of the resulting map.
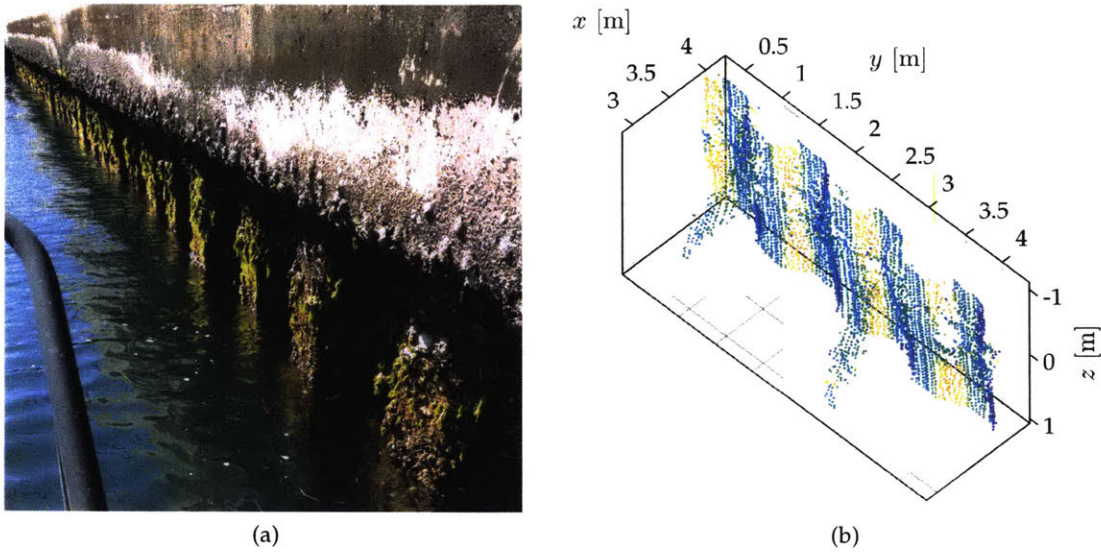
(a)   (b)

Figure 4-8: Sea wall close up: (a) photograph, looking northwest, and (b) sample submap. The seawall is located on the south side of pier 9, US Naval Base San Diego (32° 40′ 08.4″ N, 117° 07′ 10.7″ W). It features a trapezoidal corrugation pattern, with a depth of 0.2m and a pitch of 1m (approximately), and is heavily biofouled. The submap, spanning around $8m^2$ of the eastern section of seawall, clearly shows the corrugations in the wall, as well as two objects leaning against it.
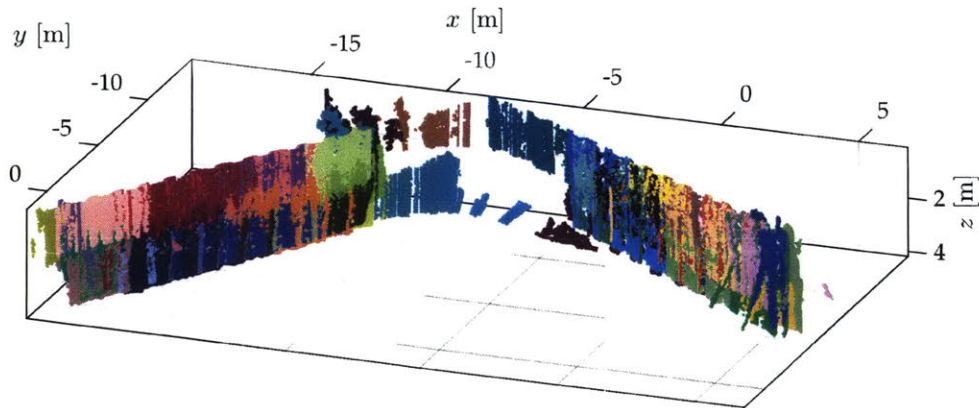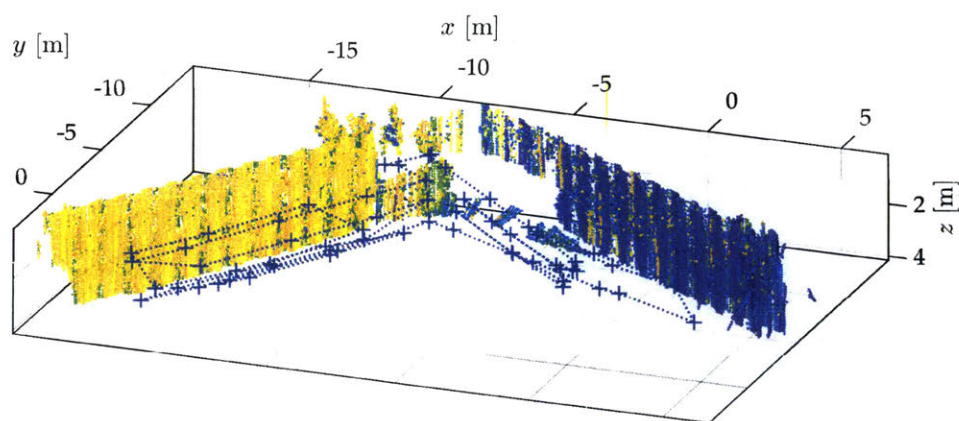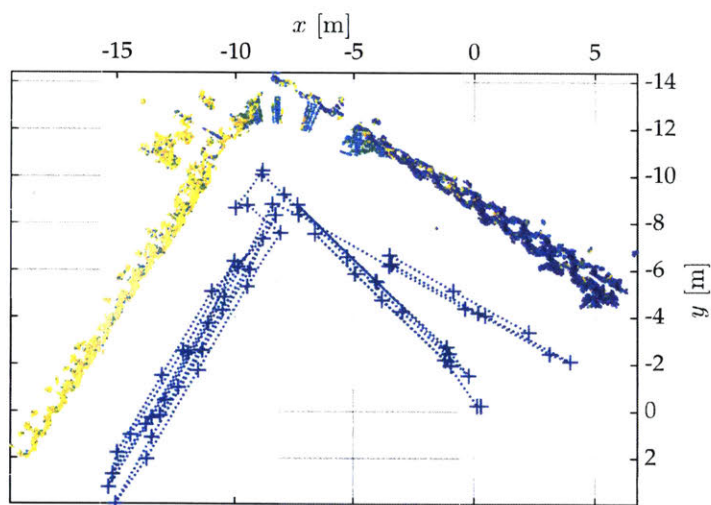

Figure 4-9: Sea wall data set - submap coverage. The data set comprises a total of 60 submaps, each spanning approximately 20 seconds, for a total of 150 platform poses.
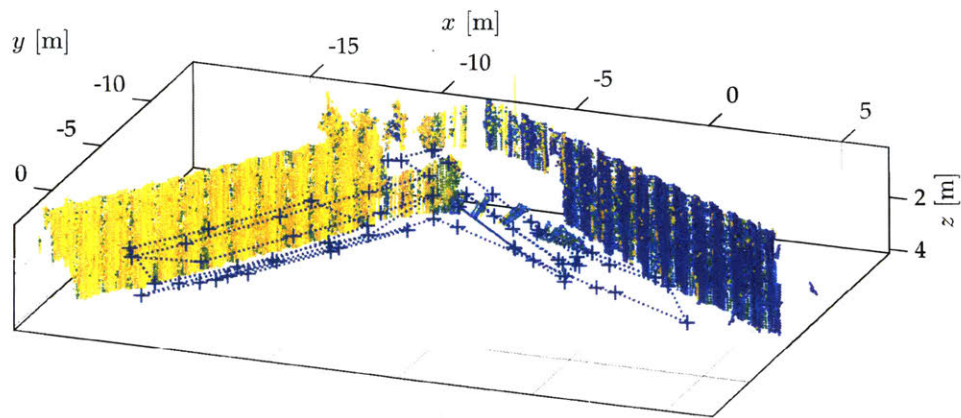
(a) Perspective

(b) Plan view

Figure 4-10: Sea wall data set - odometry estimate, colored by the x-component of the normal vectors. Submap reference positions are shown as blue crosses, connected by dashed lines. The effects of drift in the position estimate are noticeable on the bottom figure, where there is a clear misalignment between different passes over the same seawall section.
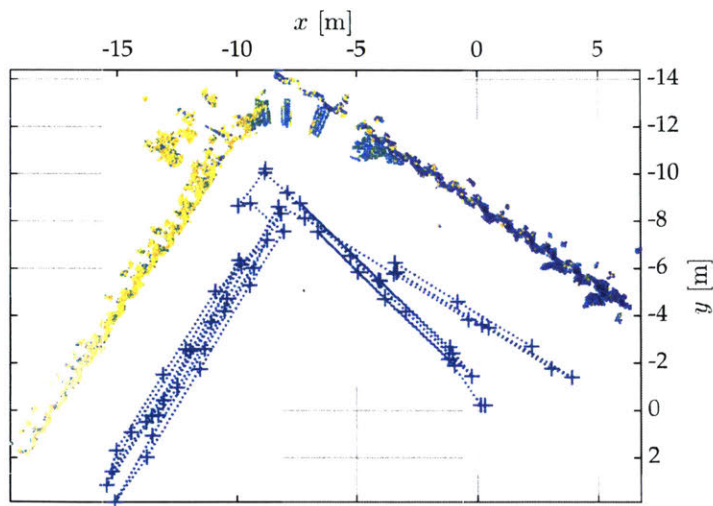
(a) Perspective

(b) Plan view

Figure 4-11: Sea wall data set - SLAM estimate, colored by the x-component of the normal vectors. Submap reference positions are shown as blue crosses, connected by dashed lines. The effects of drift visible in Figure 4-10 have been mitigated, and the multiple passes over that section of the sea wall have been successfully registered.

Chapter 5

Dense, Sonar-based Simultaneous Localization and Mapping

5.1 Introduction

In this chapter we take a different approach to the mapping problem, with the aim of combining some of the aspects from the two previous chapters: our goal is to formulate, in a unified framework, the problems of sonar segmentation (Chapter 3), pose estimation and mapping (Chapter 4), and model optimization.

We begin by revisiting our choice of map representation, which impacts how we formulate the sonar segmentation problem. Then, we address pose estimation, following a similar approach to that of chapter 4. Combining these two aspects requires tackling data association, which we accomplish by proposing a greedy yet simple incremental segmentation algorithm. Finally, we demonstrate the proposed approach on experimental data.

5.2 Related Work

As we have seen in chapter 3, multibeam sonars provide data on echo strength over range, from which range measurements must be determined. This is often accomplished by first using standard image processing operations as a pre-processing step to improve measurement accuracy, and then taking the first or strongest return above a threshold [13, 58, 61]. To address the effects of outliers typically found in sonar data (caused by acoustic phenomena such as noise, reverberation and multi-path propagation) some of the proposed methods then look at the agreement between the range measurements in consecutive beams [82]. Another group of techniques, such as those described in section 3.6, makes this relationship explicit by modeling the problem of segmenting the *full* sonar image into free and

103

occupied space using graphical models, such as Markov Random Fields [3, 105]. These *dense* methods, while more accurate, are computationally more intensive, and arguably less efficient for the purpose of obtaining a single range measurement per beam. Moreover, while the conditional dependence between neighboring sites is captured in these models, they do not leverage any information or make any explicit assumptions regarding the geometry of the scene; knowing that the range measurements correspond to points along a smooth surface, for instance, can help with the segmentation process. Such approaches are commonplace in reconstruction applications, where noisy range data is associated with some form of surface representation, often based on geometric primitives such as *planes* [109, 43], *splines* [52] or *surfels* [119]. When not assumed fixed, the pose of the range sensor (or equivalently, that of the reconstructed object) is not usually measured; instead, sensor motion (*egomotion*) is estimated through the use of Iterative Closest Point (ICP) variants from sequential range scans [119, 46]. Unfortunately, the combination of the geometry of multibeam sonars with platform motion frequently precludes the use of this family of algorithms—as platforms move perpendicular to the scanning plane to maximize coverage rate, overlap between consecutive images is eliminated. In cases where there is overlap due to in-plane motion, ICP can only provide partial (in-plane) constraints [61], and these tend to be poorly informative in the case of small fields of view. Thus, out-of-plane motion generally leads to the use of submap-based techniques, where sensor measurements and odometry are accumulated over a short time frame to produce a "virtual" sensor measurement that can be registered with previous ones and produce a relative motion estimate [82, 104, 116].

While some reconstruction methods assume drift-free sensor trajectories, avoiding the *loop closure* problem [46], others formulate it as a full SLAM problem, estimating both sensor pose and primitive location and parameters. *Planar* SLAM is a prime example of the latter, in which the proposed methods take advantage of the ubiquity of planar features in man-made environments to concurrently use as landmarks and mitigate the effect of noise in the range measurements. Most approaches estimate both the sensor pose and the parameters of planes identified from two- and three-dimensional range sensor data [109, 43] which, while noisy, is quite dense when compared to typical sonar measurements (a notable exception is the use of very sparse range data to track planar features and derive navigation constraints [70]). One particularly relevant set of techniques uses a similar approach to refine the output of a SLAM system [88, 87]: modeling range measurements as surfels, the method optimizes both sensor pose and surfel parameters. This optimization is performed iteratively: once range measurements are approximated by surfels, pairwise correspondences are then determined; once new pose and surfel estimates are available, the graph is re-built and the process continues.

5.3 Problem Statement

The problem addressed in this chapter can be described as that of estimating the position **p** of acoustic returns, as well as the pose **x** of the sensor itself. Provided an adequate scene model $\mathcal{S}$, we would also like to estimate its parameters concurrently. The inputs to our estimation problem comprise multibeam sonar scans and navigation data, the latter provided either as raw sensor measurements (e.g. from a combination of IMU, DVL, and pressure sensor), or as odometry estimates $\hat{x}_i$ from an external navigation system. Multibeam sonar scans are assumed to comprise a set of range, azimuth and elevation measurements (r_k, α_k, β_k)—these are the spherical coordinates of the scattering point in the sonar reference frame, where k indexes the beam in the scan. The choice of a scene model $\mathcal{S}$ is left, for now, as an open question.

5.4 Surface Representation

Chapter 3 described some of the methods used to derive range measurements from sonar scans, which were then used by the methods proposed in chapter 4 to build point-based representations of the scene. In these, the position of the scatterer in the world frame can be obtained by projecting the most likely range value from the sensor to the world frame (equation 2.1). What this formulation does not capture, however, is that these points are, in fact, noisy samples of some object surface: the echo does not originate from a point-like source, but rather from a surface. Given a surface representation, we can model this as a constraint between the surface and the point sample. Common candidates for discrete representations include simple geometric primitives, such as lines, planes, and *surfels* [66], as well as parametric surfaces such as splines [52].

The choice of a particular primitive (or set) is tied to the characteristics of the scene: while planes tend to be a good fit to terrestrial man-made environments, the same may not hold true for those found underwater, where curved surfaces and shapes may be more common. This is even more likely in less structured environments, such as submerged sinkholes [26] or hydrothermal vent sites.

Other representation techniques eschew geometric primitives in favor of a set of features derived from the actual scene [89], but approach the problem from a data compression perspective, taking the representation as the fixed output of a mapping system, and attempting to minimize the size of the representation while bounding the error in the reconstruction accuracy.

Given that the adequacy of a representation is intrinsically tied to the nature and characteristics of the scene, this choice entails an unavoidable trade-off between the general-

izability of the overall technique and its performance for specific types of scenes. With this in mind, we use the small-scale, spatially bounded version of planes—*surfels*—as the discrete representation of choice. Surfels are modeled by an origin $\mathbf{o} \in \mathbb{R}^3$, a unit-length surface normal $\hat{\mathbf{n}}$, and spatial support r_S. This last parameter, r_S, is key to the implicit assumption made when approximating a scene using surfels—effectively, the scene is considered to be piecewise planar at (or below) a certain characteristic scale. This scale may be determined by the scene itself, or by the end-use of the model, as there may be no need to reconstruct any detail significantly smaller than that of the features of interest. Thus, we implicitly make the assumption that, for the desired level of accuracy, the scene can be approximated by a piecewise planar set of primitives; in other words the characteristic scale of the features of interest in the scene is larger than (or at most comparable to) the spatial support r_S. For these reasons, we treat r_S not as an estimation variable but as a reconstruction parameter that is dependent upon the scene and application.
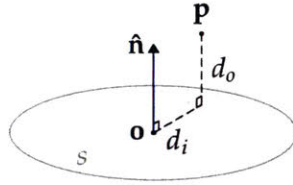

Figure 5-1: A *surface element s*, or *surfel* (also referred to as a *patch*). d_i and d_o are the *in-plane* and *out-of-plane* (or point-to-plane) distances between the point $\mathbf{p}$ and the surface element s.

5.4.1   Surfel Support

The chosen surfel parameterization has a total of 5 degrees of freedom: $\mathbf{o} \in \mathbb{R}^3$ and $\hat{\mathbf{n}} \in \mathbb{R}^3$, $\|\hat{\mathbf{n}}\|_2 = 1$ (i.e., $\hat{\mathbf{n}}$ lies on the surface of the unit sphere in $\mathbb{R}^3$). Like a plane, three points fully define a surfel; given a set of points $p_1, p_2, \ldots, p_n$, with $n \geq 3$, its parameters can be estimated using

$$\mathbf{o} = \frac{1}{n} \sum_{i=1}^{n} p_i$$

$$\hat{\mathbf{n}} = \hat{\mathbf{v}}_3$$

$$(5.1)$$

where $\hat{\mathbf{v}}_3$ is the principal component of the point set corresponding to the smallest principal value. Given a point set's principal components, $(\lambda_i, \mathbf{v}_i)$ (with $i \in \{1,2,3\}$ and $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$), the point set's *curvature c* is defined as

$$c = \frac{\lambda_3}{\sum_{i=1}^{3} \lambda_i} \tag{5.2}$$

When computed using the neighborhood of a point $\mathbf{p}$, c defines the local curvature (or *roughness*) at $\mathbf{p}$, with $c \approx 0$ (or, equivalently $\lambda_3 \ll \lambda_1, \lambda_2$) describing a locally planar region.

Measurement Model

Given a single point $\mathbf{p}_j$, and a *surfel* $s_k = \{\mathbf{o}_k, \hat{\mathbf{n}}_k\}$, we model a correspondence or assignment measurement as a combination of the *point-to-plane* and *in-plane* distances between the two, which can be written as

$$
\begin{aligned}
\mathbf{z}^s_{j,k} &= h(\mathbf{p}_j, \mathbf{s}_k) + \mathbf{v} \\
&= \begin{bmatrix} d^o_{j,k} \\ d^i_{j,k} \end{bmatrix} + \mathbf{v} \\
&= \begin{bmatrix} \sqrt{(\mathbf{n_k}^T(\mathbf{p_j} - \mathbf{o_k}))^2} \\ \sqrt{\|\mathbf{p_j} - \mathbf{o_k}\|^2 - (d^o_{j,k})^2} \end{bmatrix} + \mathbf{v}
\end{aligned}
\tag{5.3}
$$

If $\mathbf{p}_j$ is considered to be a sample of s_k, then the expected measurement value should be zero ($\mathbf{z}^s = \mathbf{0}$), with covariance $\Sigma = diag([\sigma_o^2 \ \sigma_i^2])$. The covariance associated with the point-to-plane distance measurement, σ_o^2, should be tied to the uncertainty in the association - a range measurement obtained at a grazing angle will be much more uncertain than one where the angle of incidence of the sonar beam is nearly perpendicular. The in-plane component of the distance measurement, while not strictly necessary, serves mainly to address the degeneracy issues associated with over-parameterized plane representations [43]. For this reason, and to avoid concentrating all the points around the estimate of the surfel origin, σ_i should not dominate over σ_o, and should in fact be proportional to (and greater than) the surfel support l.

5.4.2 Continuity Constraints

In assuming that, at a certain scale of interest, a scene can be considered piecewise planar and approximated by a set of surfels, we are assuming that the approximating set of surfels be somewhat evenly distributed and that their orientations should vary smoothly. To model this potential continuity constraint between neighboring surfels, we use a two-dimensional measurement comprising the pairwise point-to-plane distances between the

origin of one surfel and the plane of the other

$$\mathbf{z}^c_{j,k} = h(\mathbf{s}_j, \mathbf{s}_k) + \mathbf{v}_s$$

$$= \begin{bmatrix} d_{j,k} \\ d_{k,j} \end{bmatrix} + \mathbf{v}_s$$

$$= \begin{bmatrix} \sqrt{(\mathbf{n_j}^T(\mathbf{o_k} - \mathbf{o_j}))^2} \\ \sqrt{(\mathbf{n_k}^T(\mathbf{o_j} - \mathbf{o_k}))^2} \end{bmatrix} + \mathbf{v}_s$$

(5.4)

Given that both components of the measurement are the same, the uncertainty in the noise model for these constraints is governed by a single parameter, $\Sigma_s = \sigma_s^2 I_{2\times2}$, controlling how smooth should the resulting model be.
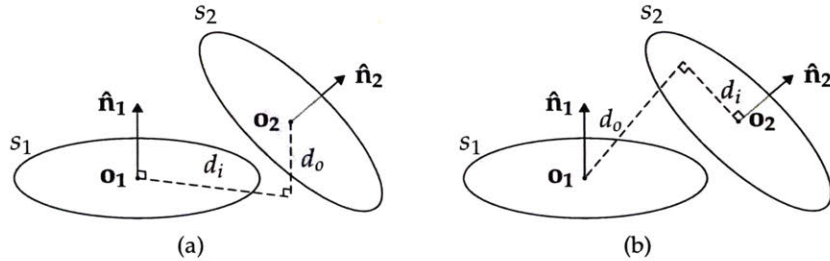

Figure 5-2: Continuity constraint: in-plane (d_i) and out-of-plane (d_o) distances between s_1 and $\mathbf{o}_2$ (a) and s_2 and $\mathbf{o}_1$ (b). The continuity constraint $\mathbf{z}^c_{m,n}$ (eq 5.4) uses the pairwise point-to-plane distances d_o to model continuity between adjacent surface elements.

5.5   Range Measurements

In the previous section we have described two measurement models associated with the chosen surfel-based representation, the last of which describes continuity/smoothness-type constraints between surfels. Equation (5.3), describing the measurement model that will effectively support the surfel-based representation, assumes a point-based estimate of the position of the sonar measurement. In order to obtain this estimate we must first revisit the sonar measurements described in the previous chapters.

Chapter 3 addressed the problem of determining a range measurement from a sonar scan, and chapter 4 addressed the projection each of these measurements along their nominal direction to form a point cloud (submap). Both of these approaches are only concerned with modeling the measurements along the sonar beam's main axis (i.e. *range*), and do not account for measurement uncertainty, particularly in the two remaining directions—

azimuth (α), and elevation (β)[1]. The uncertainty associated with these axes stems from the non-zero beam width of a sonar—considering a somewhat idealized beam (whose beam pattern is a rectangular window), the return could have originated anywhere in the ensonified cone. To model this aspect of the measurement, we consider the azimuth and elevation components to be independent and treat $p(\alpha)$ and $p(\beta)$ as static, approximated by a bivariate normal with σ_α and σ_β equal to the width of the main lobe in the respective direction (usually on the order of $1°$). Under these assumptions, the measurement model for the sonar is

$$
\begin{aligned}
\mathbf{z}_i^r &= h(\mathbf{x}_j^s, \mathbf{p}_k) + \mathbf{v}_i^r \\
&= [r\ \alpha\ \beta]^T + \mathbf{v}_i^r \\
&= \begin{bmatrix} \|\mathbf{x}_j^s - \mathbf{p}_k\|_2 \\ \arctan(\frac{p_y - y}{p_x - x}) \\ \arccos(p_z - z, r) \end{bmatrix} + \mathbf{v}_i^r
\end{aligned}
\tag{5.5}
$$

The measurement noise covariance, $\Sigma = diag([\sigma_r^2\ \sigma_\alpha^2\ \sigma_\beta^2]^T)$, is obtained from uncertainty in the range estimate (σ_r) and main lobe width in the azimuth (σ_α) and elevation (σ_β) directions. For a typical multibeam profiling sonar, as determined in Chapter 3, these correspond to $0.01m$, $1.0°$, and $1.0°$, respectively.
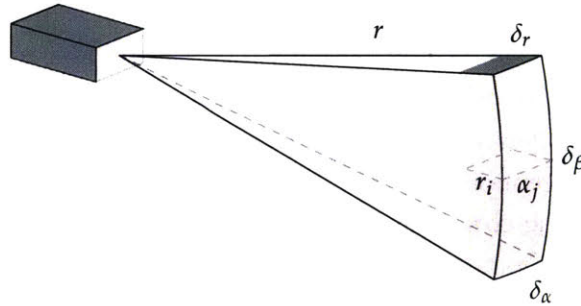

Figure 5-3: Nominal volume corresponding to a pixel in a sonar scan: δ_r is the length of the range bin, δ_α and δ_β are the beam widths in the azimuth and elevation directions, respectively.

Figure 5-4 illustrates the factor graph for the measurement models described above. In it, there are three types of variables: platform poses $\mathbf{x}_i$, location of the echoes in space, $\mathbf{p}_{i,j}$ (where i and k index the originating pose and beam, respectively), and finally, the surfels themselves, s_k. Similarly to the technique described in the previous chapter, platform poses are connected in a "chain" with relative horizontal odometry measurements, $\mathbf{z}_{i-1,i}^o$. Each

---

[1]Here we employ a spherical coordinate frame, as is typically the case when working with sonar systems.

platform pose also has an absolute measurement $\mathbf{z}_i^a$ comprising the depth and attitude of the platform. Unlike the submap-based techniques of chapter 4, platform poses are no longer marginalized to base poses; instead, the platform pose is estimated for *every* sonar scan. Each range measurement $\mathbf{z}_{i,j}^r$ relates the platform pose to the position of the scatterer which, in turn, may be associated with a surfel through a surface sample constraint $\mathbf{z}_{i,j,k}^s$. Finally, surfels may be associated between themselves through a continuity/smoothness constraint $\mathbf{z}_{k,l}^c$.

5.6   Data Association

Having described the variables and measurement models associated with the chosen map representation (eq. 5.3 and 5.4), as well as the measurement models for the point variables that support it (eq. 5.5), we now turn to the standing data association problem of how to assign points to surfels.

Starting from a set of scans and associated odometry, which we use to spatially register the range measurements and obtain an initial point cloud, we would like to derive a segmented cloud comprising *(i)* the pairwise assignments between points and surfels, and *(ii)* the adjacency between these patches, which will inform the instantiation of continuity constraints. Furthermore, if real-time operation is desired, the segmentation method must support incremental execution, as new, unsegmented, sensor data arrives.

From the many segmentation methods that could potentially be used to address this task [103], *Voxel Cloud Connectivity Segmentation* (VCCS) [76] stands out in particular, as it fulfills both the pairwise assignment and connectivity requirements. Still, due to the need for different clustering criteria in this application (explained in subsection 5.6.2) and incremental segmentation updates, we opt for implementing an arguably simpler (but potentially less efficient) method.

5.6.1   The Incremental Segmentation Problem

To facilitate the description of the proposed incremental segmentation approach, we begin by restating the problem more formally. We denote the set of labeled points $\mathbf{p}$ at time step $t = t_i$ as:

$$\begin{aligned}
\mathcal{P}_i &= \{\mathbf{p}\} \\
&= \{[\mathbf{x}^T \; \hat{\mathbf{n}}^T \; l \; t]^T\}
\end{aligned} \tag{5.6}$$

where $\mathbf{x} = [x \; y \; z]^T$ and $\hat{\mathbf{n}} = [n_x \; n_y \; n_z]^T$ are a point's location and unit normal vector, respectively. l and t are its label and acquisition time. Here we use the term *label* as shorthand for the patch to which a point is assigned, setting $l = 1$ for point assigned to patch

s_1, and $l = 0$ for unassigned points. $\mathcal{P}_i^L$ denotes the subset of points assigned to patch L:

$$\mathcal{P}_i^L = \{\mathbf{p} \in \mathcal{P}_i : l = L\} \tag{5.7}$$

Similarly, we define the set of *seed points*[2], containing exactly one point per label, as $\mathcal{S}$ (with $\mathcal{S} \subset \mathcal{P}$). To keep track of the adjacency between patches, we use the graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$, where $\mathcal{E}$ is the set of undirected edges (i, j) connecting patches s_i and s_j. Finally, we define a *comparison operator* $C(\mathbf{p}_i, \mathbf{p}_j)$, which equals true when $\mathbf{p}_i, \mathbf{p}_j$ are *similar* [110].

Starting from a (partially) labeled point set at step $i - 1$, $\mathcal{P}_{i-1}$, and the graph $\mathcal{G}_{i-1} = (\mathcal{S}_{i-1}, \mathcal{E}_{i-1})$, at step i we are given a set $\mathcal{Q}_i$ of time-stamped points $\mathbf{p} = [\mathbf{x}^T \ t]^T$. For each of these points, we would like to estimate both its normal $\hat{\mathbf{n}}$ and label l, so that we can update both $\mathcal{P}_{i-1}$ and $\mathcal{G}_{i-1}$.

5.6.2 Lossy Incremental Segmentation

Given the new set of unlabeled points $\mathcal{Q}_i$, the first step in the algorithm is to estimate their unit normals. This is accomplished by first determining, for each point in $\mathcal{Q}_i$, the set of neighboring points in both space and time, and then estimating its normal using that set (through techniques such as PCA).

Once unit normals have been estimated for $\mathcal{Q}_i$, we can proceed with an initial pairing between the new points and existing seeds in $\mathcal{S}_{i-1}$: for each point $\mathbf{q}$ in $\mathcal{Q}_i$, we search for the nearest point in $\mathcal{S}_{i-1}$, $\mathbf{s}_j$ in a neighborhood of $\mathbf{q}$, defined either by a search radius r_S or its k nearest neighbors. If the two points are similar, then $C(\mathbf{q}, \mathbf{s}_j)$ is *true* and we assign j as $\mathbf{q}$'s label; if not, then we repeat the comparison with the remaining points in the search neighborhood, in increasing order of distance until a similar point. If none are found, the point remains unsegmented for now. The point pairing process is summarized in algorithm 2, which, despite the nested loops, is fast enough for real-time usage, particularly if *k-d trees* are used to search for the k-nearest neighbors in $\mathcal{S}_{i-1}$.

Depending on the definition of the comparison operator, only a small subset of the points in $\mathcal{Q}_i$ will be paired with existing seed points; all others will remain unlabeled at this point. The next step is then to generate new seed points from this unlabeled subset, $\mathcal{Q}_i^0$. This is accomplished by sub-sampling $\mathcal{Q}_i^0$ at a resolution of r_{seed}, which we set to be equal to twice the spatial support r_S. While several techniques can be used for sub-sampling, by performing this step using a voxel grid and requiring each voxel to contain a minimum number of points (n_{min}) in order to generate a seed point, some level of outlier rejection can be achieved, at the expense of not segmenting sparsely mapped areas. The seed point generation step is summarized in algorithm 3. Once new seed points $\mathcal{S}'$ have

---

[2]Each of these can be thought of, for now, as loosely corresponding to a surfel's origin $\mathbf{o}$ and unit normal $\mathbf{n}$.

Algorithm 2: Point pairing

Input: seed point set $\mathcal{S}$, unlabeled point set $\mathcal{P}$, search radius r_S or number of
      nearest neighbors k.
Output: (partially) labeled point set $\mathcal{P}'$

1 **begin**
2   **foreach** $\mathbf{q}_j \in \mathcal{Q}_i$ **do**
3       $\mathcal{L} \leftarrow \varnothing$
4       $l_q \leftarrow 0$ ▷ Unsegmented as default ($l = 0$)
5   **foreach** $\mathbf{s}_k \in \mathcal{N}_S(\mathbf{q})$ **do**
6   **if** $C(\mathbf{q}_j, \mathbf{s}_k)$ **then**
7             $l_q \leftarrow k$
8             $\mathcal{L} \leftarrow \mathcal{L} \cup \{(j, k)\}$ ▷ Add to the list of new pairs
9   break   ▷ Move to the next point in $\mathcal{Q}_i$
10   **end**
11   **end**
12   **end**
13   **return** $\mathcal{Q}_i$, $\mathcal{L}$
14 **end**

---

been computed, they are added to the seed point set ($\mathcal{S}_i = \mathcal{S}_{i-1} \cup \mathcal{S}'$), and the adjacency graph $\mathcal{G}$ is updated, determining the adjacency between new seed points, and between these and existing seed points. This is achieved by looking, for each new seed point $\mathbf{s}'$, for similar existing seed points in its neighborhood, $\mathbf{s} \in \mathcal{N}_{S_i}(\mathbf{s}')$: if $C(\mathbf{s}', \mathbf{s})$, $\mathbf{s}' \neq \mathbf{s}$, then the two are considered adjacent and the edge $(\mathbf{s}', \mathbf{s})$ is added to $\mathcal{E}$. This process, summarized in algorithm 4, is very similar to the point pairing process (algorithm 2), with the label assignment step replaced by the addition of a new edge to $\mathcal{E}'$.

Similarly, pairing points with seeds is also performed using a greedy approach: for each point $\mathbf{q} \in \mathcal{Q}_i^0$, we iterate from the closest to farthest seed point within a search radius, and set its label to that of the *first* similar seed point, i.e., $C(\mathbf{p}_j, \mathbf{s}_k) = 1 \implies l(\mathbf{q}_j) = l(\mathbf{s}_k)$.

Thus far, we have let the comparison operator $C(\mathbf{p}, \mathbf{q})$ remain undefined to keep the segmentation process as general as possible. Since we are trying to approximate the scene by a set of small planar patches, two points should be considered similar if their normals are aligned, *i.e.*, if $\hat{\mathbf{n}}_j \cdot \hat{\mathbf{n}}_k \geq \cos(\alpha_{max})$, where α_{max} is the maximum allowable angle between the two normals. Due to the issue of drift in the sensor position estimate, it may happen that the odometry-based estimates of two distant points $\mathbf{p}_j$ and $\mathbf{p}_k$ end up in close proximity. To avoid this scenario, two points should only be considered similar if the time span between their acquisitions is short (in other words, they must also be close in space *and* time) so that the associated uncertainty is minimal, and the patch is sufficiently self-

---

Algorithm 3: Incremental seed point generation

Input: Unlabeled new point set, $\mathcal{Q}_i^0$, grid size d
Output: New seed points, $\mathcal{S}'$

1 **begin**
2     $\mathcal{S}' \leftarrow \emptyset$;
3     $\mathcal{L}_{\cdot,\cdot,\cdot} \leftarrow \emptyset$; $\triangleright$
4   **foreach** $\mathbf{q} \in \mathcal{Q}_i$ **do**
5        $(i,j,k) \leftarrow \left(\left\lfloor \frac{q_x}{d} \right\rfloor, \left\lfloor \frac{q_y}{d} \right\rfloor, \left\lfloor \frac{q_z}{d} \right\rfloor \right)$;
6        $\mathcal{L}(i,j,k) \leftarrow \mathcal{L}(i,j,k) \cup \{\mathbf{q}\}$
7   **end**
8   **foreach** $(i,j,k) : \mathcal{L}_{i,j,k} \neq \emptyset$ **do**
9        $n \leftarrow \#\mathcal{L}(i,j,k)$;
10   **if** $n \geq n_{min}$ **then**
11          $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{ \frac{1}{n} \sum_{\mathbf{q} \in \mathcal{L}(i,j,k)} (\mathbf{q}) \}$
12   **end**
13   **end**
14   **return** $\mathcal{S}'$
15 **end**

---

consistent. Thus, we define the comparison operator as

$$C(\mathbf{p}_i, \mathbf{p}_j) = \left(|t_i - t_j| < \delta t_{max} \right) \wedge \left(\hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_j \geq \cos \alpha_{max} \right) \tag{5.8}$$

Under this definition, the comparison operator $C(\cdot, \cdot)$ can also be interpreted as a threshold-based decision rule on some distance function between two points.

Given a current point set $\mathcal{P}_{i-1}$, graph $\mathcal{G}_{i-1}$ and new point set $\mathcal{Q}_i$, the three processes above are repeated until there are no new pairings, at which point the updated (and potentially non-exhaustively labeled) point set $\mathcal{P}_i$ and graph $\mathcal{G}_i$ can be used to update the factor graph through the addition of new patch nodes, point and patch associations, and continuity constraints. The need for iterative segmentation within each incremental update is driven by the seed point generation mechanism: if, for example, a voxel contains two sets of dissimilar points, acquired around t_i and t_k, just one of these sets will be segmented after one step, as only one seed point will have been chosen from each voxel at a given iteration. By repeating the process of seed point generation, we minimize the number of unlabeled points. The incremental segmentation algorithm is summarized in algorithm 5. We use the term *lossy* due to the fact that the segmentation is not complete, *i.e.*, some of the points will not be assigned to patches, remaining unlabeled.

Each incremental update of the proposed algorithm has some resemblances to *k*-means

---

Algorithm 4: Seed point adjacency update

Input: Current graph $\mathcal{G}_{i-1} = (\mathcal{S}_{i-1}, \mathcal{E}_{i-1})$, new seed points $\mathcal{S}'$
Output: Updated graph $\mathcal{G}_i = (\mathcal{S}_i, \mathcal{E}_i)$

1 **begin**
2    $\mathcal{E}' \leftarrow \varnothing$
3    $\mathcal{S}_i \leftarrow \mathcal{S}_{i-1} \cup \mathcal{S}'$
4   **foreach** $s_j \in \mathcal{S}'$ **do**
5   **foreach** $s_k \in \mathcal{N}_{\mathcal{S}_i}(s')$ **do**
6   **if** $C(s_j', s_k)$ and $s_j \neq s_k$ **then**
7             $\mathcal{E}' \leftarrow \mathcal{E}' \cup \{(s_j, s_k)\}$
8   **end**
9   **end**
10   **end**
11   $\mathcal{E}_i \leftarrow \mathcal{E}_{i-1} \cup \mathcal{E}'$
12   **return** $\mathcal{G}_i = (\mathcal{S}_i, \mathcal{E}_i)$
13 **end**

---

clustering, particularly how it relies on "greedy" assignments of a point to a cluster center through the use of a distance function—these correspond to the seed points and to the binary-valued comparison function $C(\cdot, \cdot)$, respectively. The main limitations preventing the use of standard k-means clustering techniques stem from the unknown number of clusters and its inability to guarantee an upper bound on the distance between a point and its cluster center.

Model Update

After a set of new sonar points $\mathcal{Q}_i$ has been registered and the segmentation has been updated, the factor graph model can be revised. This entails the addition of new variable nodes for poses, points, and patches, as well as the corresponding factors, as illustrated in figures 5-4 and 5-5a. The set of new, segment points $\mathbf{p}_i$, surfels s_k and edges (s_k, s_l) can be obtained either directly from algorithm 5, or from the set difference between the segmentation state $(\mathcal{P}, \mathcal{G})$ at steps $i - 1$ and i. As described in the previous chapter, the odometry "chain" is extended with every new sonar scan, through the addition of the corresponding pose node $\mathbf{x}_i$, relative odometry measurement $\mathbf{z}_{i-1,i}^o$ (eq. 4.19), and absolute depth and attitude measurements, z_i^d and $\mathbf{z}_i^q$ (eqs. 4.21 and 4.20, respectively), as shown in figure 5-4a. For each measurement in scan i, a point node $p_{i,j}$ is added, together with a range, azimuth, and elevation measurement $\mathbf{z}_{i,j}^r$ (eq. 5.5) where j indexes the beam in that scan (figure 5-4b). Once the point variables are present, the segmentation output, $\mathcal{P}_i$ and $\mathcal{G}_i$, is used to inform the addition of both the patch variables s_k, and the surface sample

---

Algorithm 5: Lossy Incremental Segmentation

Input: Current segmented point set $\mathcal{P}_{i-1}$, seed point graph $\mathcal{G}_{i-1} = (\mathcal{S}_{i-1}, \mathcal{E}_{i-1})$, and new point set $\mathcal{Q}_i$

Output: Updated segmented point set $\mathcal{P}_i$ and seed point graph $\mathcal{G}_i = (\mathcal{S}_i, \mathcal{E}_i)$,

1 **begin**
2      $\mathcal{Q}_i \leftarrow$ pairPoints($\mathcal{Q}_i$, $\mathcal{S}_{i-1}$)
3   **repeat**
4          $\mathcal{S}' \leftarrow$ generateSeedPoints($\mathcal{Q}_i^0$)
5   **if** #$\mathcal{S}'$=0 **then**
6   break
7   **end**
8          $\mathcal{G}_{i-1}$ $\mathcal{Q}_i$, $\mathcal{L} \leftarrow$ pairPoints($\mathcal{Q}_i^0$, $\mathcal{S}'$)
9   **until** #$\mathcal{L} = 0$
10      $\mathcal{P}_i = \mathcal{P}_{i-1} \bigcup \mathcal{Q}_i$
11      $\mathcal{G}_i \leftarrow \mathcal{G}_{i-1}$
12   **return** $\mathcal{P}_i, \mathcal{G}_i = (\mathcal{S}_i, \mathcal{E}_i)$
13 **end**

---

measurements $\mathbf{z}_{i,j,k}^s$ (eq. 5.3) that support them (fig. 5-4c). To keep with the assumptions on the piece-wise planarity of the scene, only patches with curvature estimates below a maximum value (c_{max}) are added to the factor graph; all others are discarded from $\mathcal{G}$. At this point, the continuity constraints $\mathbf{z}_{k,l}^c$, derived from the new edges added to $\mathcal{E}_i$ in the segmentation update, are instantiated as measurements between patches (fig. 5-5a).

5.7 Loop Closures

The scene model described in the previous section aims at improving the accuracy of the reconstruction at a small scale: by assigning most points to patches and constraining the latter to vary smoothly with respect to their neighbors, we hope to produce a smooth scene model that is less sensitive to noise and uncertainty in the sonar measurements. While these constraints help addressing inaccuracies at small spatial and temporal scales, they are not able to mitigate the effects of long-term drift in the platform's pose estimate; to do so we must extend our approach so that it can detect loop closures and integrate these in the model.

The scene model described in section 5.4 comprises a set of patches, each of which is supported by a number of points corresponding to the positions of sonar measurements—the former are definitely richer features than the latter, but still they do not contain sufficient information to allow for unambiguous association of two patches. While a richer scene representation would facilitate the loop closure detection task, it must be compatible
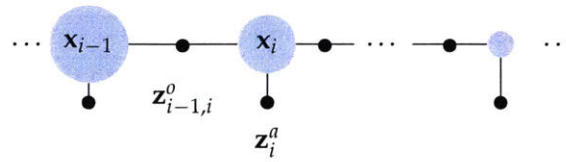
(a) Step 1 - odometry "chain", comprising pose nodes x_i, relative odometry $z^o_{i-1,i}$, and absolute ZPR measurements z^a_i.
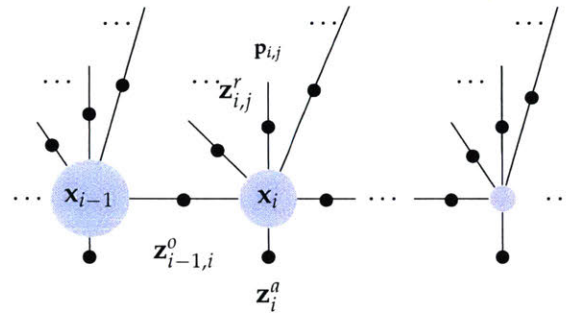

(b) Step 2 - sonar return position nodes $p_{i,j}$ and range, angle and elevation measurements $z^r_{i,j}$.
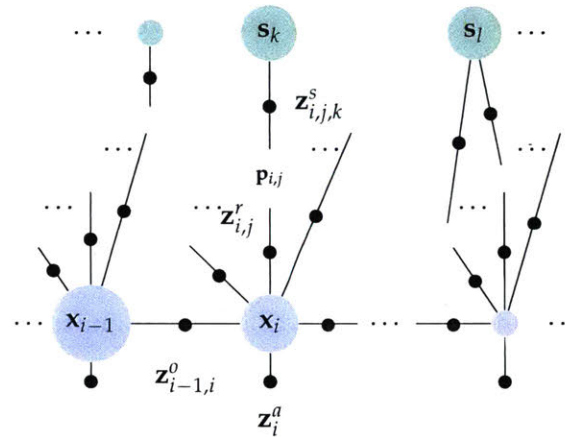

(c) Step 3 - scene model, comprising patch nodes s_k and surface sample measurements $z^s_{i,j,k}$.

Figure 5-4: Factor graph model for dense reconstruction. Each vehicle pose node x_i, in blue, is connected to the previous pose by a relative odometry measurement $z^o_{i-1,i}$ (eq. 4.19). The node x_i is also constrained by the absolute depth, pitch, and roll measurement z^a_i (eqs. 4.21 and 4.20). The range measurement $z^r_{i,j}$ (eq. 5.5) connects the vehicle pose x_i to the corresponding scatterer position, p_j, represented by the gray node. A scatterer may be associated with a surface element s_k, in which case it is constrained by the measurement $z^s_{j,k}$ (eq. 5.3). . For clarity purposes, only one of each type of factor has been labeled in this figure.
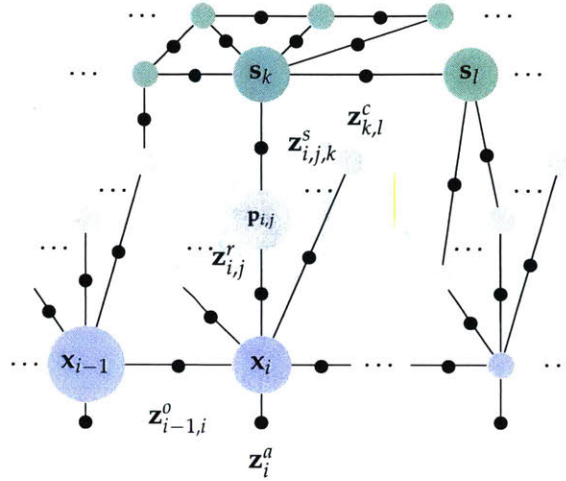
(a) Step 4 - Continuity constraints between patches, $\mathbf{z}_{k,l}^c$.
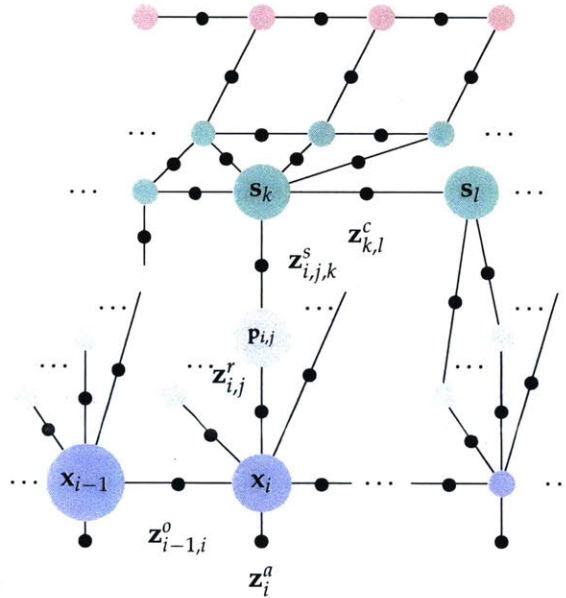
(b) Step 5 - Match constraints between patches, $\mathbf{z}_{m,n}^c$.

Figure 5-5: Factor graph model for dense reconstruction (continued from figure 5-4). Continuity constraints $\mathbf{z}_{k,l}^c$ (eq. 5.4) between adjacent surface elements enforce the continuity/smoothness of the scene (figure 5-5a). Finally, match constraints $\mathbf{z}_{m,n}^m$ (eq. 5.13) are derived from pairwise registration of partition elements and connect surfels in one element for which a corresponding surfel in the other element was found (figure 5-5b).

(a) Initial point cloud

(b) After 1 iteration

(c) After 2 iterations
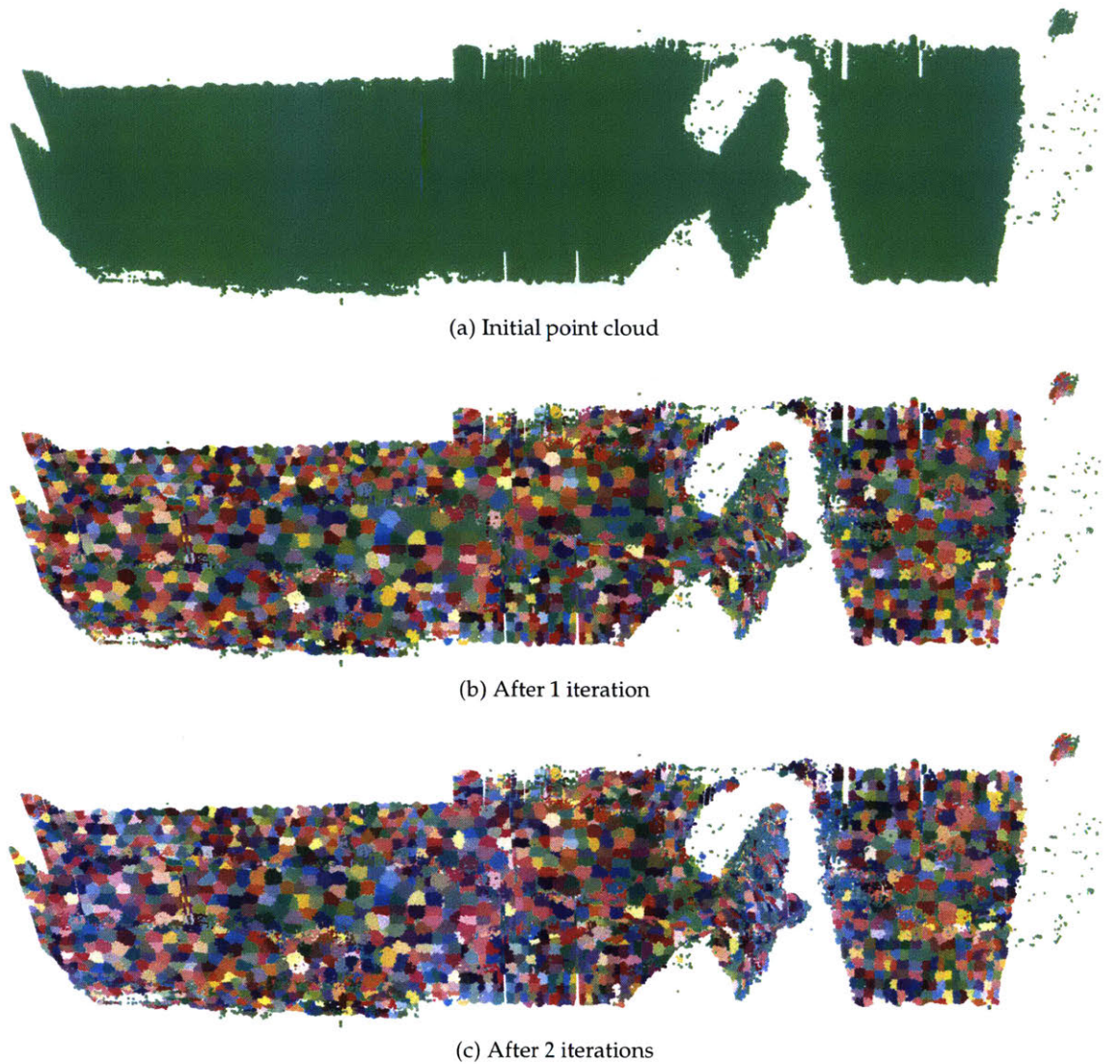
Figure 5-6: Lossy Incremental Segmentation - sample results. This figure shows just the first two iterations of the algorithm, which terminates in 7 iterations. The resulting segmentation yields 4104 segments, out of which 3886 are considered planar. The final number of unsegmented points is 8669, out of a total of 331436—a segmentation loss of less than 3%.

with the characteristics of both the chosen sensing modality and the scenes of interest for a particular application. Given our initial assumption of piece-wise planarity of the scene at or below a given scale of interest, the scene should not contain rich or unique information at that scale. This, combined with the nature of the sensor measurements supporting the surface elements (simple range measurements) and the comparatively sparse coverage density the sensor provides would make an approach using rich feature descriptors rather challenging, if not scene-dependent. For these reasons, we instead opt to consider *sets* of low-level features (surface elements in this case) as higher-level features.

Note that this is similar to what was proposed in the previous chapter—as pairs of sonar scans could not be associated unambiguously, we instead chose to accumulate them into submaps and use these to derive loop closures. Unlike that technique, however, we are no longer forced to consider all the poses and points within a submap as fixed with respect to the base pose.

5.7.1 Graph Partitioning

By choosing to consider sets of patches as high-level features we must tackle the problem of how to create the sets in the first place. We consider this as a (graph) *partitioning* problem, as we would want to divide S into non-overlapping subsets that we can use to derive loop closure constraints in order to mitigate long-term drift.

Fixed-length Partitions

The approach proposed in chapter 4 partitioned the trajectory along fixed-length segments—by choosing the length (time span) appropriately (equation 2.16), it ensured an upper bound on the uncertainty of each pose in the segment with respect to the chosen base pose. Consequently, the map produced from that segment would also have an upper bound on the uncertainty associated with the position of the sensor measurements—in other words, it would be self-consistent. This approach can be readily transferred to the dense formulation by partitioning the patch set S according to the time associated with each patch s, t_s (i.e., the timestamp of its seed point). We define the ith element of the partition as the set

$$\mathbb{P}_i = \{s \in S : t_s \in [t_i, t_{i+1})\} \tag{5.9}$$

where the length of the interval $[t_i, t_{i+1})$ is set using the same criteria as in subsection 4.4.3.

Dynamic Partitions

Partitions can also be derived by leveraging the information contained in the graph $\mathcal{G}$, in particularly, the adjacency information contained in $\mathcal{E}$. As we have seen in chapter 4, the geometric "richness" of a submap is a critical aspect to its ability to provide an informative loop closure— two nearly-planar submaps, for example, will result in an underconstrained registration. To avoid these situations, we can use the diversity of normals in a set of patches as a proxy measure for the qualitative geometric "richness" of that set. To build this pseudo-measure of normal diversity, we use an objective function over two surfels s_k and s_l that increases with the angle between their normals, θ:

$$
\begin{aligned}
h(s_k, s_c) &= 1 - |\hat{\mathbf{n}}_k \cdot \hat{\mathbf{n}}_l| \\
&= 1 - |\cos\theta|
\end{aligned}
\tag{5.10}
$$

This function is maximum (1) when $\hat{\mathbf{n}}_c$ and $\hat{\mathbf{n}}_k$ are perpendicular and minimum (0) when they are parallel (we consider the absolute value of $\cos\theta$ as some of the normals may be flipped). For a set of surfels $\mathbb{P} = \{s_k\}$ and a candidate surfel s_c, the gain in adding s_c to $\mathbb{P}$ is:

$$
g_c = \min_k \left(1 - |\hat{\mathbf{n}}_k \cdot \hat{\mathbf{n}}_c|\right)
\tag{5.11}
$$

Thus, if we are to choose from a set of candidates $\{s_i\}$, the surfel with the greatest minimum angle difference with respect to surfels in $\mathbb{P}$ provides the greatest increase in normal orientation diversity. The optimal candidate is then:

$$
\begin{aligned}
s_i^\star : i &= \arg\max_i g_i \\
&= \arg\max_i \min_k 1 - |\hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_k|
\end{aligned}
\tag{5.12}
$$

To use this objective function to derive a subset $\mathbb{P}$, we begin by identifying the connected components in the non-partitioned subset of $\mathcal{G}$. For each component, we start by selecting a random node as the initial element of the new subset, $\mathbb{P} = s$. The subset is expanded by considering all the nodes adjacent to $\mathbb{P}$ and selecting those for which, were they to be included, the timespan criteria would continue to hold. From these candidates, the new element of $\mathbb{P}$ is selected according to (5.12). This process is repeated until there are no more candidates, at which point $\mathbb{P}$ is removed from the set and a new random node is selected.

5.7.2 Pairwise Registration

Once the subsets $\mathbb{P}_i$ have been determined, the next step is to determine and evaluate potential pairwise associations. Despite the surfel-based representation, we leverage the

same techniques as in chapter 4, namely, the use of ICP-type algorithms to register subsets. Unlike the previous chapter, however, the point clouds used in the registration process are no longer comprised of sonar returns; instead, we take the seed points of each surfel in the subset – these provide the significant advantages of being supported by *several* sonar measurements, decreasing the sensitivity to outliers and eliminating the need for the pre-processing step. Moreover, as a result of the seed generation process in the incremental segmentation algorithm, the points in patch-based point are nearly uniformly spaced. Combined, these properties make these clouds well-suited for ICP-type algorithms.

As described in subsection 4.6.2, ICP-type techniques iterate between estimating the correspondences between two point sets and the rigid-body transformation between the corresponding subsets of points. Whereas in submap-based techniques it is this transformation estimate that is used as a relative pose measurement between the associated base poses, in the proposed formulation we are interested in the correspondences instead—as the two point clouds are obtained from the initial estimates of the origins of the patch nodes in each of the two subsets, $\mathbb{P}_i$ and $\mathbb{P}_j$, the pairwise correspondences estimated by ICP equate to matches between the surfels from which the points were obtained.

5.7.3 Surfel Match

To incorporate the pairwise match information in our model, we first need to model it as a measurement dependent on the two matching surfels. As a correspondence signifies that the two patches are the same—the two span the same area of the scene, and have the same orientation—the measurement can be modeled as a difference between their parameters, with an expected value of zero ($\mathbf{z} = \mathbf{0}$). Like other measurements, this difference is subject to uncertainty modeled as a zero-mean multivariate normal distribution. The covariance of this distribution, Σ_m, is mostly dependent on the confidence associated with the correspondence. The measurement model is then

$$
\begin{aligned}
\mathbf{z}_{j,k}^m &= h_M(s_j, s_k) + \mathbf{v}_s^m \\
&= \begin{bmatrix} \mathbf{o}_j - \mathbf{o}_k \\ \hat{\mathbf{n}}_j - \hat{\mathbf{n}}_k \end{bmatrix} + \mathbf{v}_s \\
&= [\delta_x \; \delta_y \; \delta_z \; \delta_u \; \delta_v \; \delta_w]^T + \mathbf{v}_s
\end{aligned}
\tag{5.13}
$$

The pairwise match measurements between surfel variables are illustrated in figure 5-5b, where binary measurement factors connect patch nodes in different subsets, indicated by the red and green colors.

5.8   Experimental Results

In this section we present experimental results for the methods described in this chapter. We begin by demonstrating the incremental segmentation algorithm described in subsection 5.6.2 on a segment of experimental data. Then, we proceed by experimentally validating the overall reconstruction method on a longer inspection segment, spanning thousands of poses.

5.8.1   Data sets

The data set used for experimental evaluation of the proposed method is a segment taken from the same *SS Curtiss* inspection session as that of chapter 3. The chosen segment comprises a total of 6000 sonar scans, acquired over approximately 15 minutes. During this period, the platform performed five constant-depth legs of a vertical lawn-mower pattern, moving laterally at around $0.2 - 0.25 \ m.s^{-1}$, with the sonar pointed towards the ship, and configured so that the scanning plane was parallel with the ship's cross-section, to maximize coverage rate. The chosen lawn-mower pattern, shown in figure 5-9 was mostly parallel to the ship's longitudinal axis, with a depth increments of $1 \ m$ and a horizontal span of about $30 \ m$.

5.8.2   Parameters

Based on the sonar properties, we set $\sigma_\alpha = 0.3°$ and $\sigma_\beta = 1.0°$, and use a conservative $\sigma_r = 0.02m$ for range [6]. For odometry measurements, we let $\Sigma_a = diag([0.1^2 \ (1°)^2 \ (1°)^2]^T)$ and $\Sigma_o = \delta_i t \cdot diag(1 \times 10^{-6}[9 \ 9 \ 1]^T)$ for absolute and relative odometry measurements, respectively, where $\delta t_i = t_i - t_{i-1}$. For this particular segment, the platform moves laterally (perpendicular to the sonar plane) with $|v| \approx 0.2 \ m.s^{-1}$, so we let $r_S = 0.15 \ m$ and $\delta t_{max} = 2 \ s$, and require a minimum of 10 points to generate a seed.

5.8.3   Reconstruction Results

We implemented the methods described in sections 5.6.2 through 5.7.1, leveraging iSAM [44] for factor graph optimization, and PCL [91] for point cloud processing and visualization.

The results of the algorithm on the ship hull data set are shown in figures 5-7 through 5-14. Out of the initial 367,187 points, the resulting reconstruction generated 6792 seed points, from which 6792 patches, connected by 39923 edges, were added to the model. These are shown in figures 5-12 and 5-13, respectively. Only 3796 points were left unsegmented—a loss of approximately 1%.
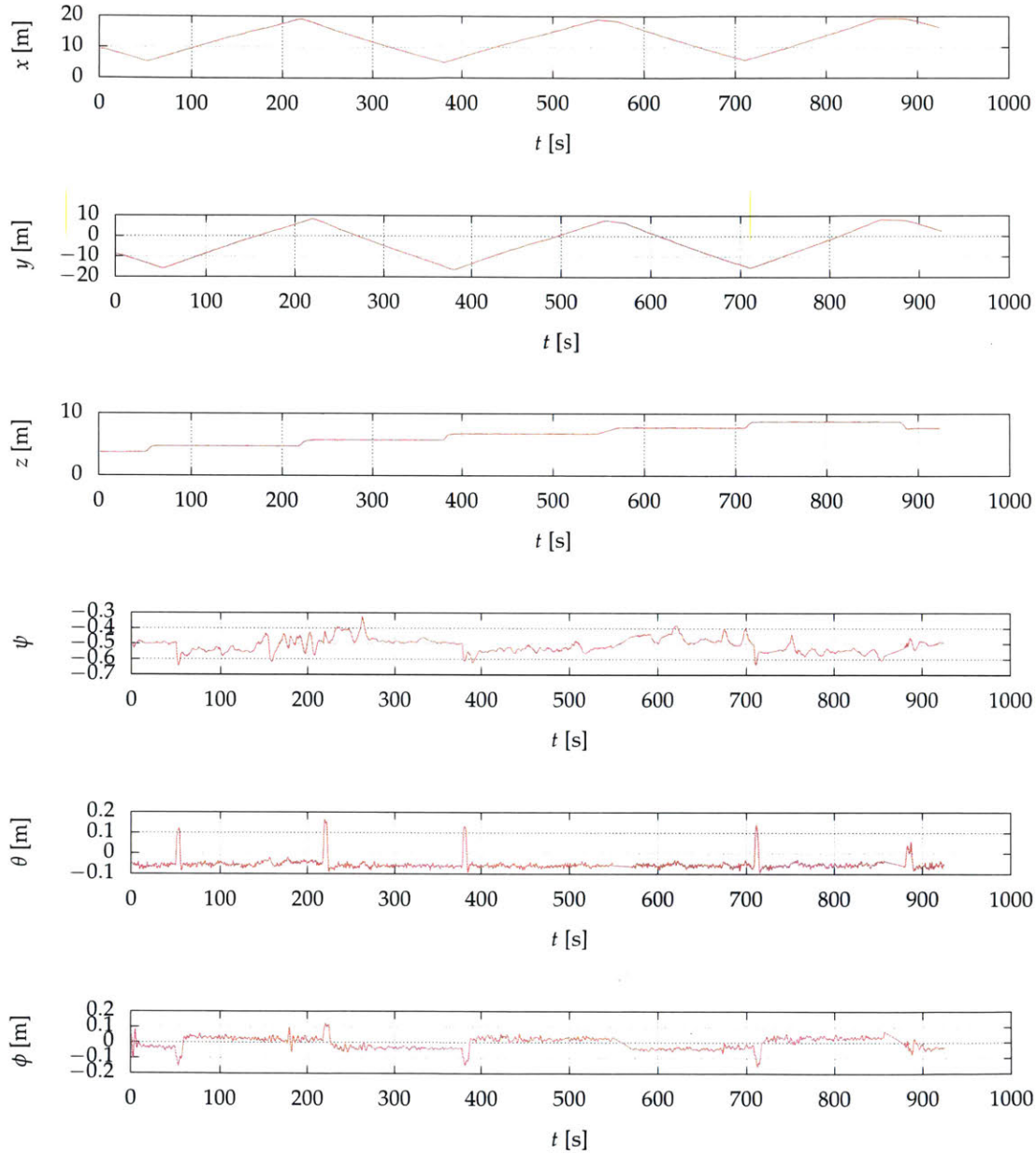
Figure 5-7: Odometry-based pose estimates for the full trajectory. While the platform is stable underway, there are some visible transients when adjusting depth, namely in both roll (ϕ) and pitch (θ). As a result of the platform's single lateral (y) thruster being off-axis, and the lack of roll authority, there is a noticeable, direction-dependent roll offset on the order of a few degrees when in motion. Depth changes also produce short-term deviations in pitch of approximately 10-15°.
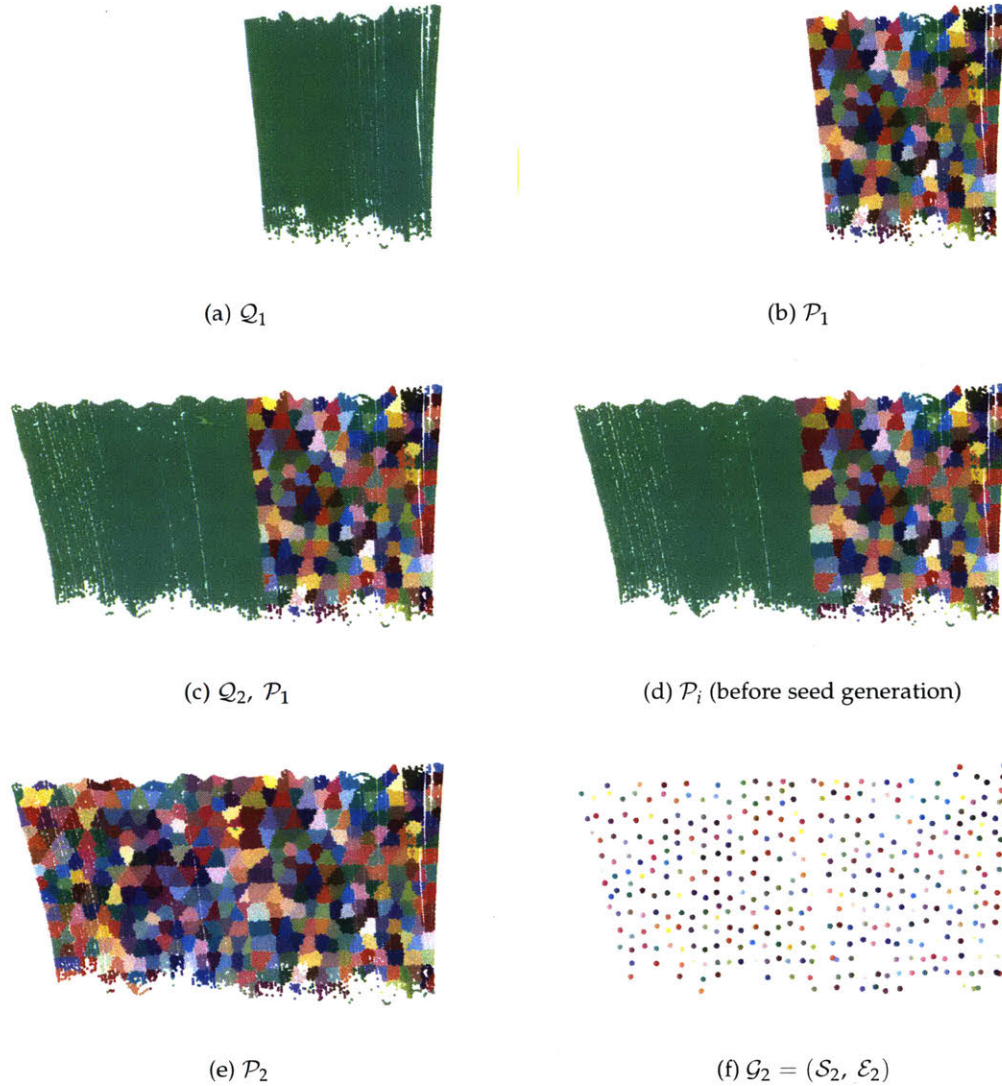
(a) $\mathcal{Q}_1$

(b) $\mathcal{P}_1$

(c) $\mathcal{Q}_2$, $\mathcal{P}_1$

(d) $\mathcal{P}_i$ (before seed generation)

(e) $\mathcal{P}_2$

(f) $\mathcal{G}_2 = (\mathcal{S}_2, \mathcal{E}_2)$

Figure 5-8: Lossy incremental segmentation - sample results illustrating initialization and update steps in a nearly-planar scene. When the first set of points $\mathcal{Q}_1$ is received (a), $\mathcal{P}_0 = \mathcal{S}_0 = \varnothing$ and the seed pairing step is skipped. Once new seeds are generated and the points are paired (b), we obtain $\mathcal{P}_1$ and $\mathcal{S}_1$. A second set of points $\mathcal{Q}_2$ is received (c) and the segmentation must be updated: new points are paired with existing seeds (d) before the iterative seed generation/point pairing process is restarted, resulting in the updated segmentation, $\mathcal{P}_2$ (e), $\mathcal{G}_2 = (\mathcal{S}_2, \mathcal{E}_2)$. Platform motion is horizontal, right-to-left, and $\mathcal{Q}_i$ comprises 150 scans.

Figures 5-9 through 5-11 show the odometry- and SLAM-based results for the ship hull inspection data set. The effects of drift are clearly visible in the odometry-based map: the multiple horizontal passes over the hull produce a set continuous surfaces, but these do not overlap. This can be seen in the smooth gradient in the top half of figures 5-9 and 5-10, as well as on the different coloration in the top half of figure 5-11. These effects are successfully mitigated in the SLAM solution: all three views of the resulting map show the cloud segments from the five different passes overlapping. This difference is also visible on the rudder, which appears as two distinct segments in the odometry solution, which are brought closer using the SLAM estimates.

Mapping Performance

To quantify the performance of the proposed mapping technique, we will focus on a small segment of the inspection session, comprising the first pass over the running gear (screw rudder) of the ship, and spanning 750 scans acquired over approximately two minutes (roughly the interval from 120 to 240 seconds in figure 5-7). As this segment does not contain overlap in coverage, the techniques presented in chapter 4 will not be able to provide an improvement upon the odometry-based map estimate.

Figure 5-16 illustrates the uncertainty associated with the odometry- and SLAM-based map estimates by coloring each point has been colored according to to its uncertainty. This is achieved by computing, for every point, the square root of the *Frobenius* norm of its covariance matrix Σ,

$$\sqrt{\|\Sigma\|_F} = (tr(\Sigma^T \Sigma))^{\frac{1}{4}}$$

which we consider to be a scalar approximation of the total uncertainty in the estimate.

In figure 5-16a, the vehicle trajectory starts forward of the screw and ends aft of the rudder. The non-zero uncertainty in the first scans is dominated by both the vertical uncertainty caused by the limited accuracy of the pressure sensor, and by the angular uncertainty in the sonar range measurements. These effects are particularly particularly visible in areas that have been registered from a longer range, such as the starboard blades and the section of the hull below the shaft. As the vehicle moves, the uncertainty in the horizontal position estimate from the aided INS grows and starts to dominate over the other two sources (which remain relatively constant).

Figure 5-16b illustrates the uncertainty associated with the a SLAM-based map estimate. Here we can see a significant reduction in uncertainty (by a factor of 2-5) thanks to the use of a scene model. The growth in uncertainty remains noticeable, but this is to be expected as no loop closures have been derived or used for this segment.

Finally, uncertainty values for both the odometry- and SLAM-based estimates for longer
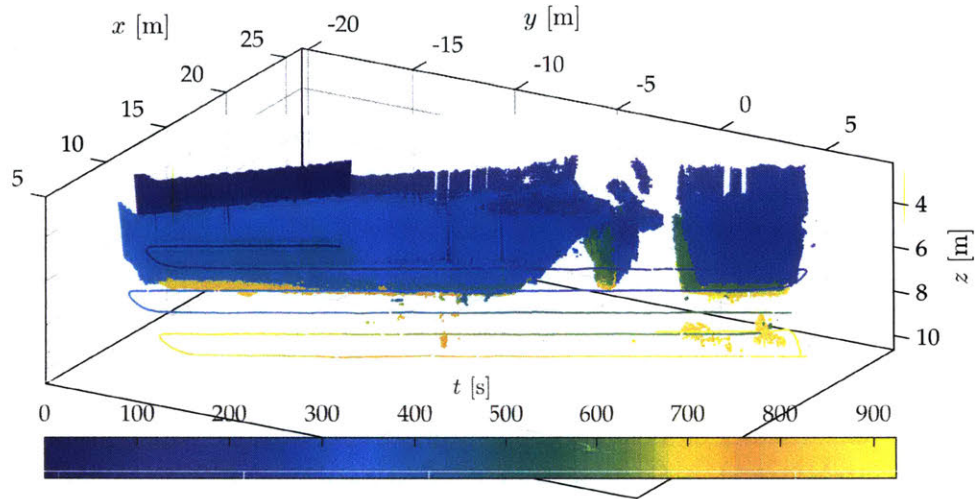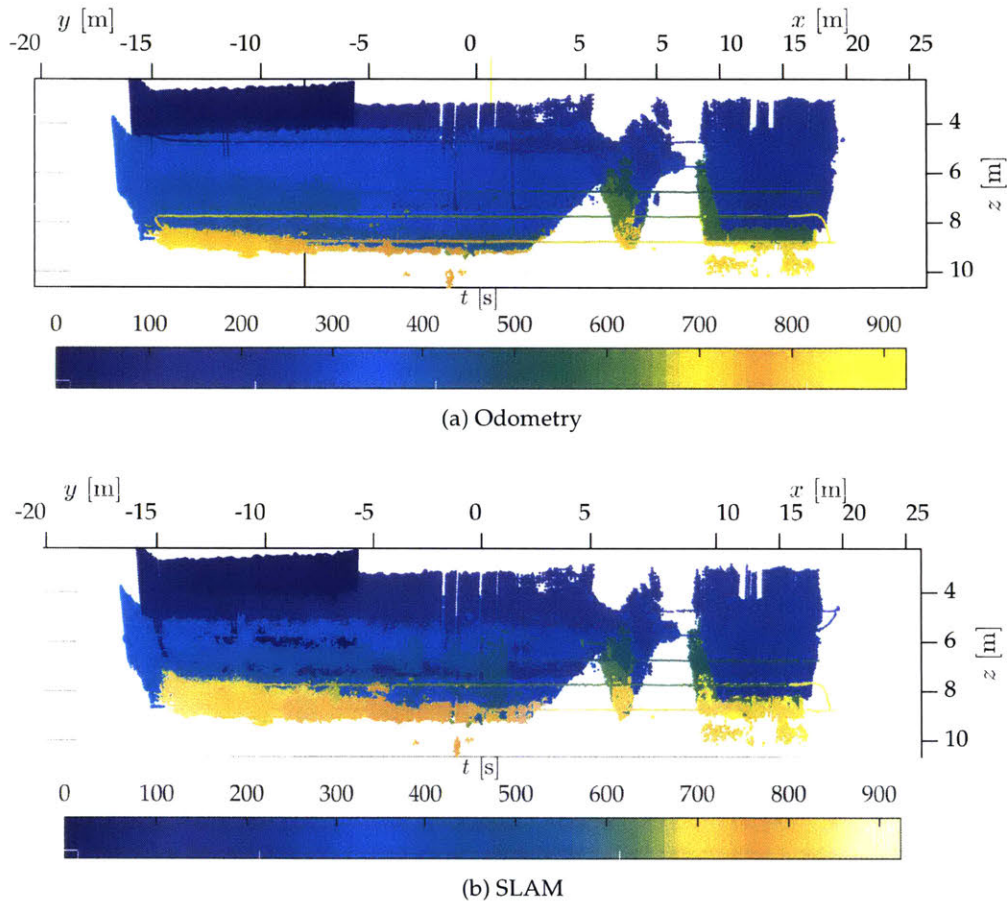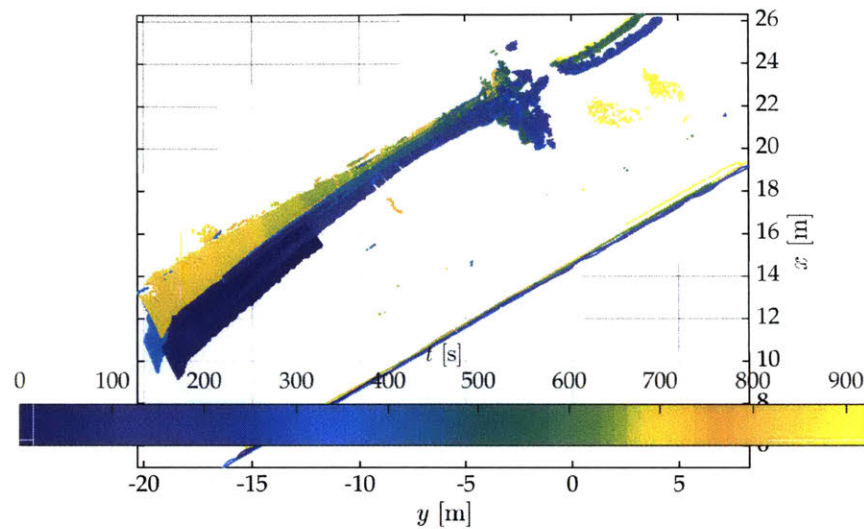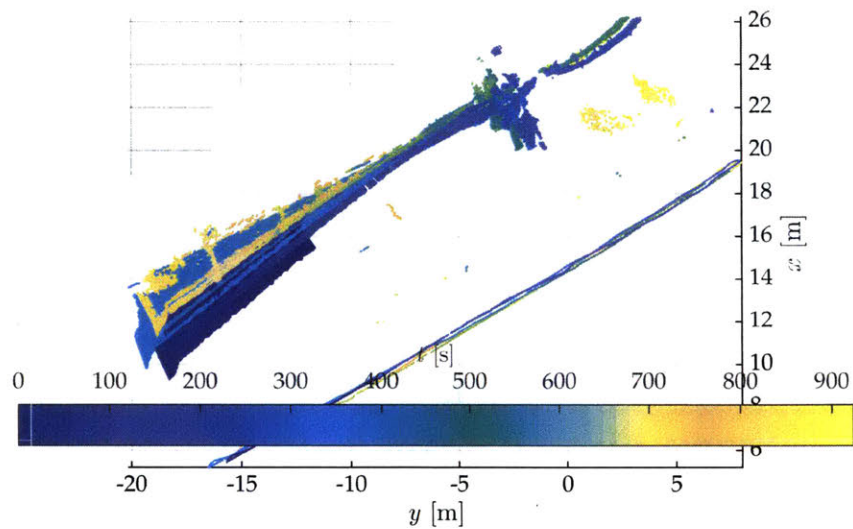
(a) Odometry
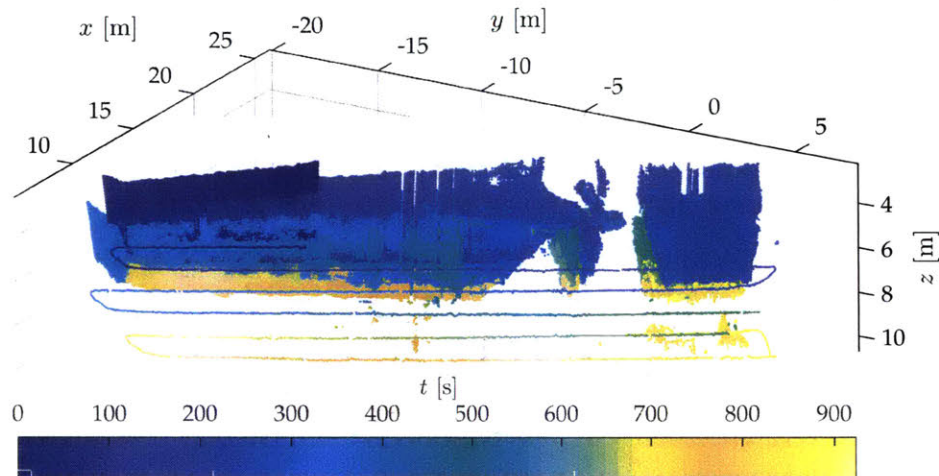
(b) SLAM

Figure 5-9: Ship hull data set 2018-03-14.00: perspective view, colored by acquisition time. This set spans 6000 poses acquired in approximately 15 minutes. During that time, the platform executed 5 full constant-depth legs of a vertical lawn-mower pattern port of the ship. The effects of odometric error are visible in the top image: the different passes over the hull do not intersect, as drift has resulted in a small port-to-starboard offset between measurements from different passes. This has been successfully mitigated in the SLAM solution (bottom), where all the passes intersect correctly.
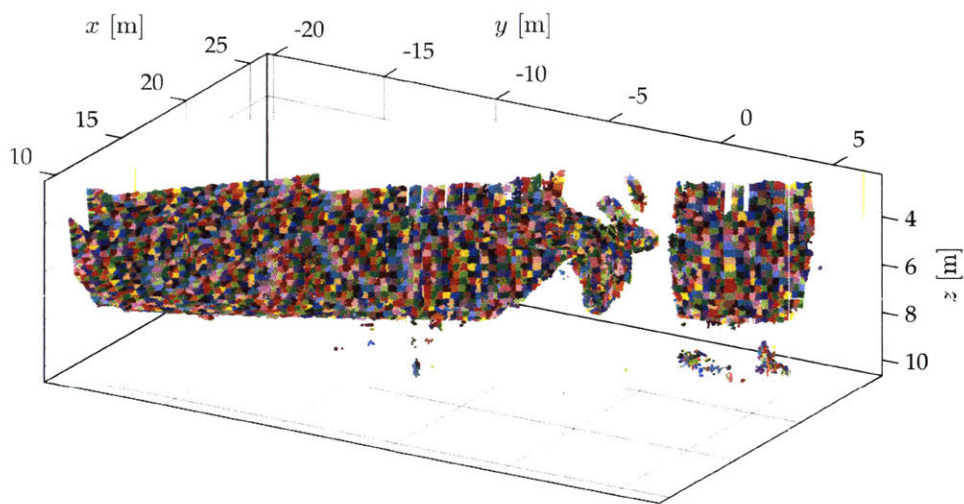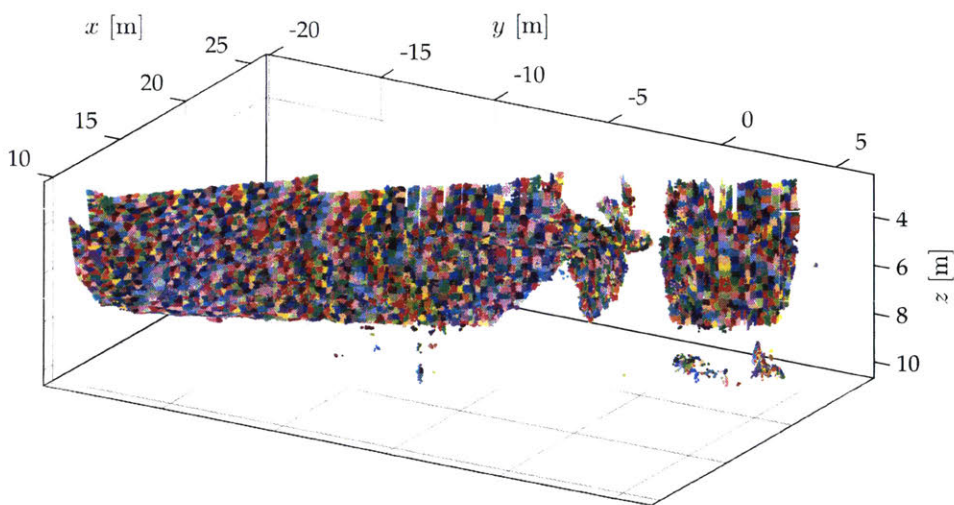
(a) Odometry

(b) SLAM

Figure 5-10: Ship hull data set 2018-03-14.00: profile view, colored by acquisition time. This set spans 6000 poses acquired in approximately 15 minutes. During that time, the platform executed 5 full constant-depth legs of a vertical lawn-mower pattern port of the ship. The effects of odometric error are visible in the top image: the different passes over the hull do not intersect, as drift has resulted in a small port-to-starboard offset between measurements from different passes. This has been successfully mitigated in the SLAM solution (bottom), where all the passes intersect correctly.

(a) Odometry

(b) SLAM

Figure 5-11: Ship hull data set 2018-03-14.00: plan view, colored by acquisition time. This set spans 6000 poses acquired in approximately 15 minutes. During that time, the platform executed 5 full constant-depth legs of a vertical lawn-mower pattern port of the ship. The effects of odometric error are visible in the top image: the different passes over the hull do not intersect, as drift has resulted in a small port-to-starboard offset between measurements from different passes. This has been successfully mitigated in the SLAM solution (bottom), where all the passes intersect correctly. The effects of drift can also be seen on the rudder (top right of each image), which appears as two offset instances in the odometric plot. While this drift-induced artifact has not completely eliminated in the SLAM solution, the effect is significantly less pronounced.

(a) Odometry

(b) SLAM

Figure 5-12: Ship hull data set 2018-03-14.00: segmented point cloud $\mathcal{P}$. The difference between the odometry- and the SLAM-based solutions can be seen in the details of the segmented point cloud: while the odometry-based cloud appears closer to a regular mosaic, the updated pose, point, and surfel estimates in the SLAM solution have resulted in noticeable "distortion" of the tiles.
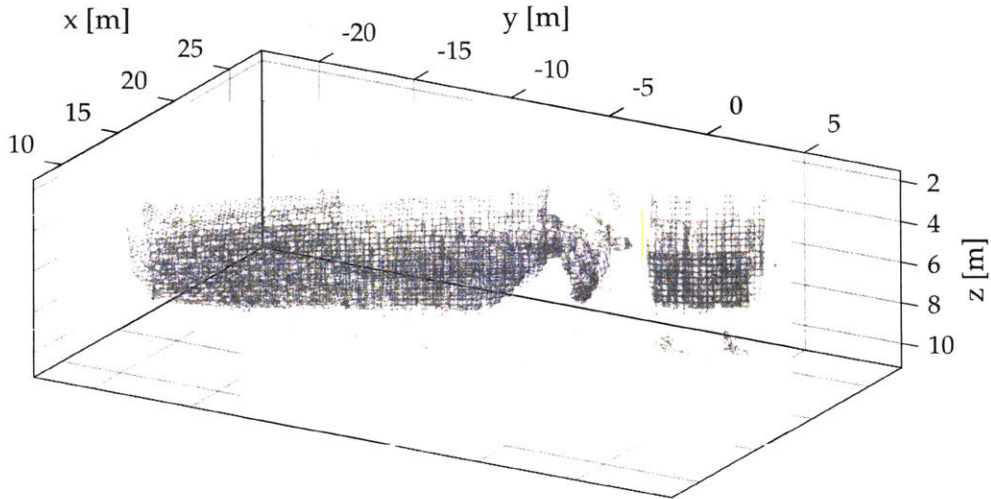
Figure 5-13: Ship hull data set 2018-03-14.00 - seed graph $\mathcal{S}$. Out of the 240201 sonar measurements, the segmentation process resulted in a total of 238198 labeled points (a loss of less than 1%), and 4583 seed points, from which 4021 patches planar patches were selected. These were supported by 214801 points, and connected by 16763 edges.
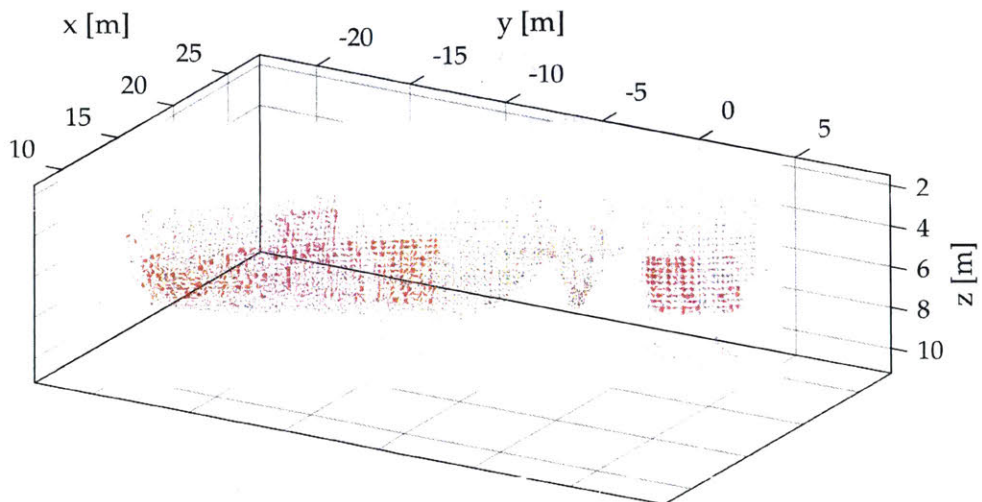

Figure 5-14: Ship hull data set 2018-03-14.00 - pairwise matches between surfels, shown as red edges connecting the seed point origins corresponding to each patch.

(a) New measurements   (b) Segmentation

(c) New measurements

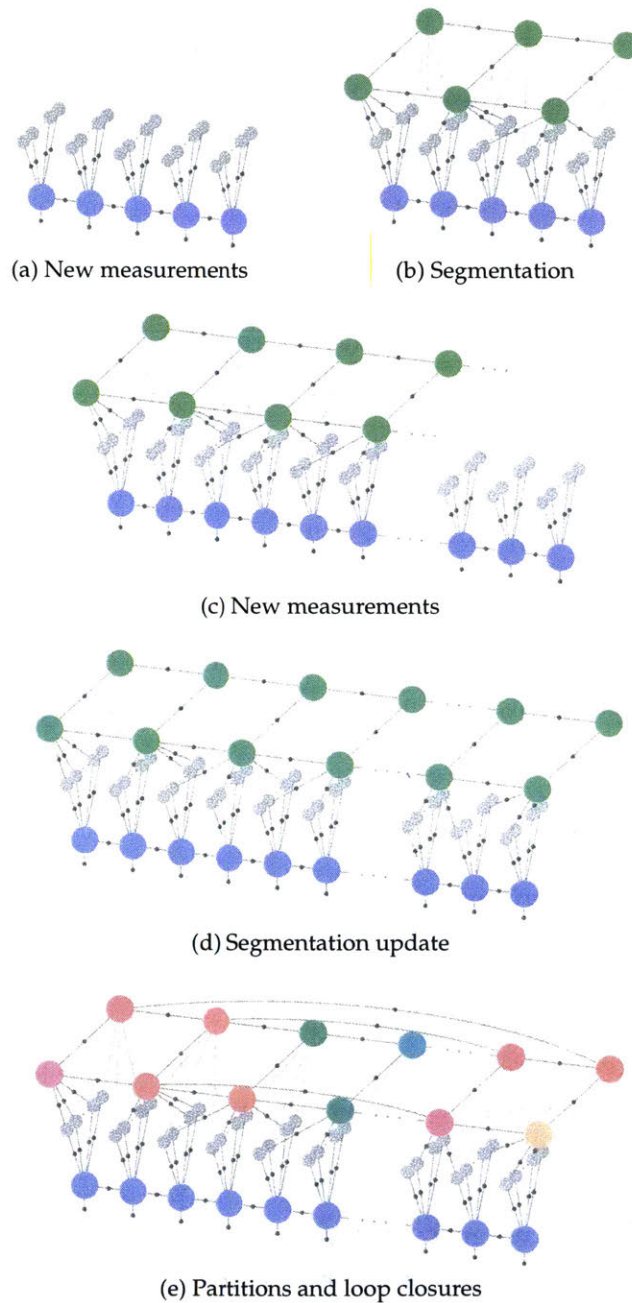(d) Segmentation update

(e) Partitions and loop closures

Figure 5-15: Dense SLAM: incremental updates: (a) once a set of measurements has been received, the model is updated with the latest poses and sonar measurements. (b) once the new data has been segmented, new surfels and supporting measurements are added. (c,d) the process is repeated once a new set of measurements arrives. (e) periodically, the partition is updated, and loop closures are instantiated as pairwise matches between surfels.

n	min	max	μ	σ
1500	0.093	0.236	0.177	0.013
3000	0.091	0.236	0.179	0.013
4500	0.091	0.237	0.182	0.013
6000	0.091	0.237	0.183	0.013

Table 5.1: Odometry-based map accuracy for different trajectory lengths (n, the number of poses). The listed values correspond to the *minimum, maximum, mean (μ)*, and *standard deviation (σ)* in the value of $\sqrt{\|\Sigma\|_F}$ for all the points in the map.

| n	min	max	μ	σ
1500	0.005	0.025	0.013	0.003
3000	0.019	0.103	0.048	0.009
4500	0.020	0.118	0.050	0.010
6000	0.002	0.008	0.005	0.001

Table 5.2: SLAM-based map accuracy for different trajectory lengths (n, the number of poses) without loop closures. The listed values correspond to the *minimum, maximum, mean (μ)*, and *standard deviation (σ)* in the value of $\sqrt{\|\Sigma\|_F}$ for all the points in the map.

segments are listed in tables 5.1 and 5.2—from these we can see that despite the lack of loop closures, the use of sensor and scene models significantly improves map accuracy.

5.9 Summary

We presented a reconstruction method that addresses the artificial separation between sensor processing, pose estimation, and model reconstruction in the scope of sonar-based mapping. Leveraging scene information to aid in sonar processing requires a shared model between these tasks—we choose one commonly used for pose estimation: factor graphs. The surfel-based, piece-wise planar approximation of a scene was proven to work experimentally. Still, it requires that some attention be paid to certain parameters, namely, the spatial support/characteristic scale r_S, and the relative weight of range and surfel sample constraints, σ_R and σ_S . In particular, σ_S is likely to be pre-determined by the sonar and scene properties, as the return signal will depend on them.

   While the current implementation is unable to produce results in real time, this is simply a consequence of the very large number of variables in the problem—as demonstrated in table 5.3, most of the computation time is spent on factor graph optimization. The number of variables in the current formulation is by far dominated by the point variables; without these, the number of variables in the seventh row of table 5.3, for example, would decrease by over an order of magnitude. Still, the proposed approach was devised with the goal of real time application, hence the emphasis on the incremental (as opposed to
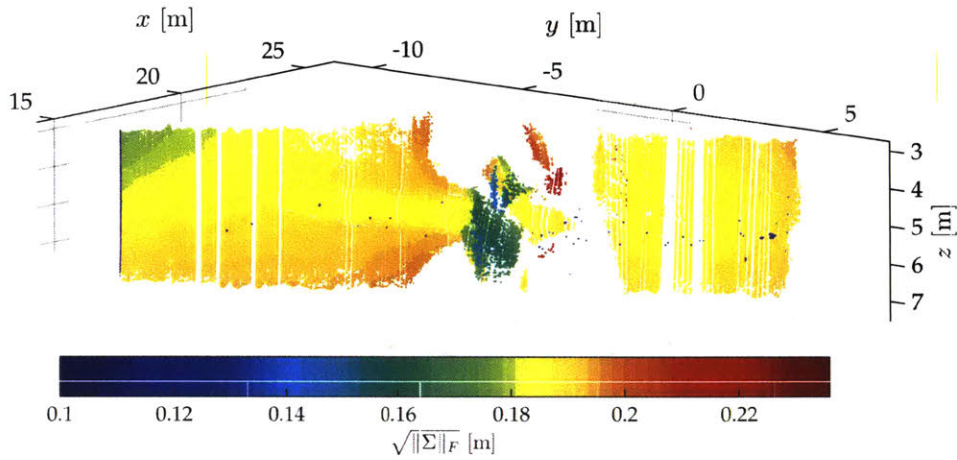
(a) Odometry-based map estimate
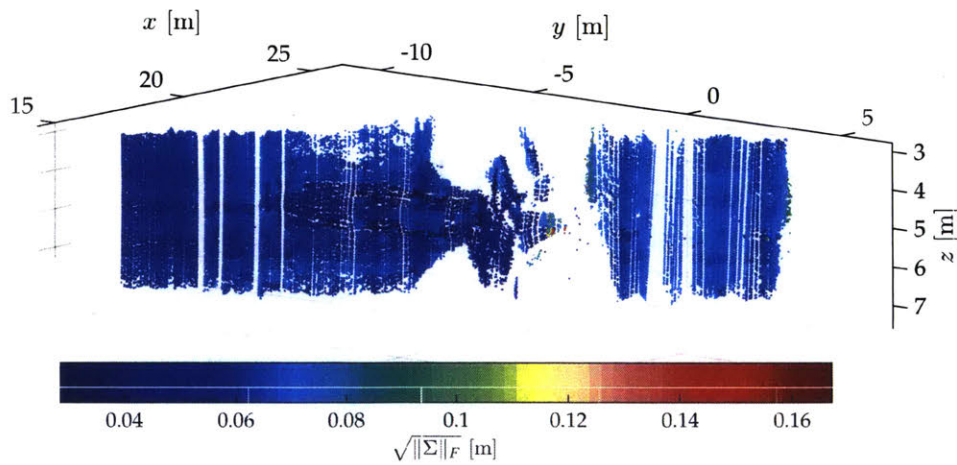
(b) SLAM-based map estimate

Figure 5-16: Map uncertainty for the odometry- and SLAM-based map estimates. The mean and standard deviation values for $\sqrt{\|\Sigma\|_F}$ are 0.186 ± 0.008m and 0.062 ± 0.009m for the odometry- and SLAM-based estimates, respectively.

batch) nature of the key processes—segmentation, partitioning, and model updates. These techniques would be just as applicable if point variables were to be dropped from the formulation.

Length	Variables			Factors				R		Iteration time					
	Poses	Points	Patches	XYH/ZPR	RAE	Adjacency	Match	Rows	Non-zero	t_0	t_1	t_2	t_3	t_4	Total
0:02	750	66677	1023	749	66677	3849	257	210669	0.013%	0:05	0:05	0:04	0:02	0:02	0:21
0:04	1500	117695	1902	1499	117695	6003	257	373497	0.007%	0:31	0:04	0:06	0:04	0:04	0:53
0:06	2250	170368	2829	2249	170368	11813	366	541578	0.005%	0:26	0:05	0:05	0:05	0:05	0:54
0:07	3000	229514	3852	2999	229514	20287	802	729654	0.005%	0:29	0:07	0:07	0:07	0:07	1:06
0:09	3750	272939	4522	3749	272939	23333	1128	868449	0.004%	0:37	0:07	0:08	0:07	0:07	1:19
0:11	4500	306878	5192	4499	306878	26327	1465	978786	0.003%	1:12	0:09	0:10	0:10	0:09	2:06
0:13	5250	342956	5857	5249	342956	31351	1562	1095510	0.003%	1:14	0:20	0:11	0:10	0:11	2:23
0:15	6000	367187	6238	5999	367187	35413	1695	1174989	0.003%	0:49	0:23	0.11	0:12	0:11	2:05

Table 5.3: Factor graph parameters and run time performance (in *hours:minutes* format) for increasingly long session segments on a commodity laptop computer (2018). The total time includes the time taken by the five sequential full optimization steps and the additional time for loading, segmenting, and partitioning the map. The observed worst-case processing time required per second of data for a full optimization step ranges between 2 and 7 seconds, precluding real-time usage of the proposed technique.

Chapter 6

Conclusion

6.1 Conclusion

We addressed the problem of sonar-based, high-resolution mapping for underwater scenes by focusing on two of the main sources of uncertainty described in chapter 2: vehicle pose estimates and sensor range measurements. Chapter 3 proposed sonar processing and segmentation techniques that aimed at increasing the accuracy of range measurements in the presence of artifacts and noise. These techniques were successfully demonstrated using real sensor data. Chapter 4 built upon these results and addressed the issue of pose estimation accuracy. To mitigate drift in the odometry-based pose estimate, we leveraged small-scale maps ("submaps") from which we derived pairwise constraints that supplemented the odometry estimates. These proved successful at mitigating drift in long (>1 hour) mapping sessions. Finally, chapter 5 described a technique aimed at addressing both range and pose estimation problems simultaneously. This approach was enabled by choosing a more complex map representation, which, where possible, made an explicit assumption about the scene: that it could be considered locally planar. Moreover, this model allowed us to represent loop closures as a semi-dense set of constraints, using the set of correspondences derived from pairwise registration of partitions instead of the estimated transform. While the computational cost of this dense mapping technique precludes its use in real-time applications, the next section lists some promising directions that may bring it closer to that goal.

6.2 Future Work

6.2.1 Assumptions, revisited

Multi-modal Distributions

At several points in this text, particularly when addressing the problems of sonar segmentation and range measurement modeling, we have made strong assumptions regarding the associated distributions, for instance: section 3.6.2 assumed at most one return per beam, and section 5.5 approximated the uncertainty in azimuth and elevation in a range measurement as a bivariate normal distribution. Neither of these assumptions are valid – multi-path propagation and other acoustic effects such as reverberation will often result in multiple copies of the transmitted signal being received by the same beam; at the same time, figure 3-14 clearly demonstrates the inherently multi-modal nature of the uncertainty in the azimuth of a range measurement. Relaxing these assumptions would thus require the use of inference techniques that support multi-modal distributions [31].

Dynamic Scenes

In many inspection applications, scenes or objects are likely to not be static; some of the more relevant scenarios include iceberg mapping, inspection of moored ships in rough weather, inspection of berthed ships while loading/unloading, and fish farm inspection. More extreme scenarios could include hull inspection while underway, in which case the design of the entire navigation system may have to be revisited, as open-sea altitudes may be beyond reach of most DVLs commonly found in inspection platforms. Such applications would necessitate revising the proposed techniques to work in a relative (moving) reference frame, or to explicitly model the rigid body motion of the body under inspection.

6.2.2 Sensor Offsets

Like the DVL and IMU, the mapping sensor payload also requires calibration to determine $_S^P T$, as lever arm effects can introduce registration errors in the order of tens of centimeters when mapping at a long range. To address the issue, we model the offset between sensor and platform as part of the estimation problem: with every new sonar measurement a new measurement is added:

$$
\begin{aligned}
\mathbf{z}_i^c &= h(\mathbf{x}_i^p, \mathbf{x}_i^s, c) + v_i^c \\
&= {}_S^P\hat{T}\, {}_S^W\hat{T}_i^{-1}\, {}_P^W\hat{T}_i + v_i^c
\end{aligned}
\tag{6.1}
$$

This measurement model describes what is essentially a "consistency check" between the platform and sensor poses and the sensor offset; if all three estimates are correct, the measurement should yield the identity transformation. Future work should attempt to validate this model. [22, 112, 120]
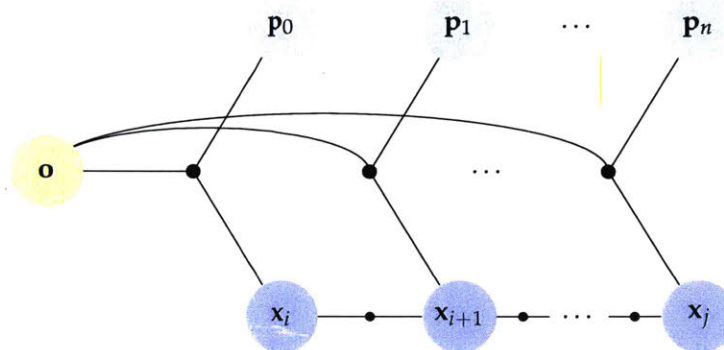

Figure 6-1: Sensor offset measurement model

6.2.3 Extensions and Improvements

Semantic Segmentation

Extending the sonar scan segmentation methods proposed in section 3.6 to provide not just the range to the scatterer but also the *type* of scatterer (e.g. "hull", "piling", or "seafloor") would prove imensely useful not only for the association of sonar measurements with geometric primitives (subsection 5.6.2), but also when partitioning higher-level scene segments (subsection 5.7.1). Section 3.6.1 points at one such direction; the parameters in the object intensity distribution are likely to be directly related to the type (and acoustic characteristics) of the object. Provided the objects in a scene yield sufficiently different distributions, we should expect estimates of these parameters to be grouped into clusters.

Hierarchical Landmarks

Hierarchical landmarks provide a more formal description of the concepts introduced by graph partition in subsection 5.7.1, and another step in the path towards unifying the pose estimation and scene reconstruction problems under the same factor graph framework. In this extension, a landmark would be described by a center pose variable, representing the landmark's position in space, and a set of "neighbor" nodes describing the geometric primitives (e.g. surfels, planes, splines); connecting each peripheral node to the center node are low-uncertainty, fully constraining factors—this concept is illustrated in figure 6-
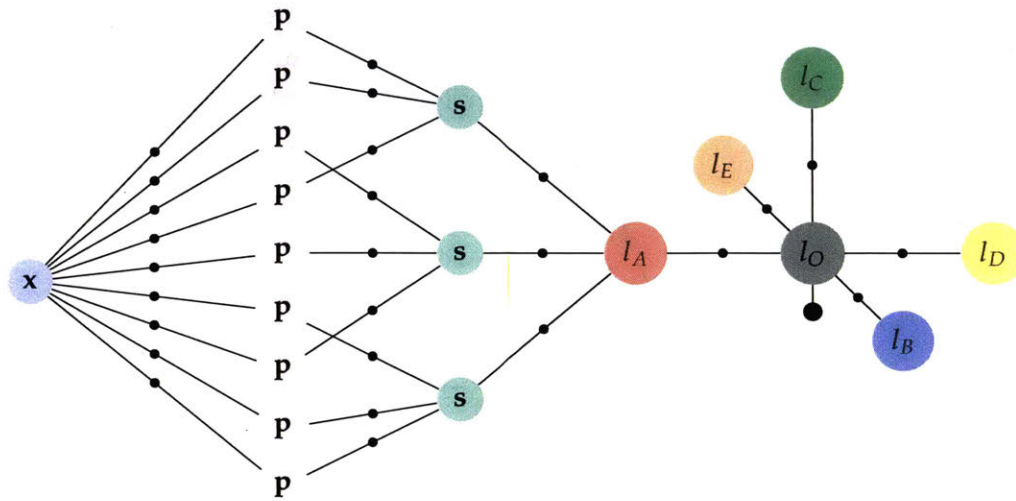
Figure 6-2: Hierarchical landmark concept - a complex object, such as a piece of subsea infrastructure, is described as a set of landmarks with tight relative constraints: the model origin l_O represents the reference center of the object, and carries a measurement prior describing its measured or estimated position in the world frame. Other features in that infrastructure block are modeled as individual landmarks—l_A through l_E. These could represent several types of features: fiducial markers for optical imaging systems, affordances for manipulators, *etc.*

2. By explicitly modeling the landmark through explicit geometric primitives we facilitate data association, as well as a greater variety of measurement models.

Sparser Formulations

One of the most significant limitations of the factor graph model described in chapter 5 is that its high dimensionality precludes its use in real-time applications. This stands in contrast with the segmentation, partitioning, and registration techniques that are used to build it, which are amenable to both incremental and batch use, and can be executed in real-time. For this reason, one of the areas warranting immediate attention is the development of techniques that reduce the size of the problem.

Implicit Point Representation   The formulation and techniques described in chapter 5 make use of range, azimuth and elevation measurements as well as point-surface correspondence measurements that rely on the explicit representation of points as estimation variables. This may prove advantageous for the purpose of deriving synthetic sonar data to use as ground truth for the actual sonar measurements, for instance. However, given that there are usually hundreds of points in a single sonar scan, these point variables are

the dominant factor in the dimensionality of the estimation problem. One of the immediate ways to address this issue is to combine the two measurements into a single constraint affecting a pose and a surfel variable, as illustrated in figure 6-3. Using this technique,
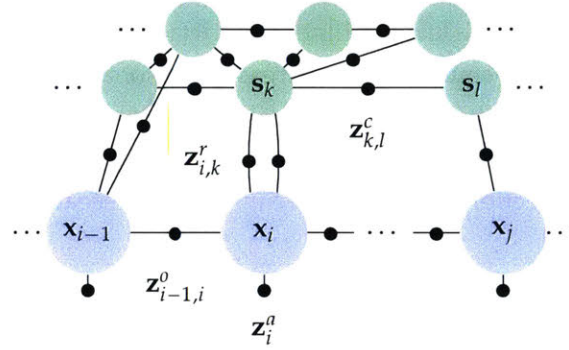

Figure 6-3: Implicit point range measurement factor, replacing the point variables and associated measurements in figure 5-4 (eqs. 5.5 and 5.3) with a single measurement (eq. 6.2).

the point variables, together with the range measurements (eq. 5.5) and surface sample constraints (eq. 5.3), would be replaced by a single combined measurement

$$
\begin{aligned}
\mathbf{z}_i^r &= h(\mathbf{x}_j^s, \mathbf{p}_k) + \mathbf{v}_i^r \\
&= \begin{bmatrix} \|\mathbf{x}_j^s - \mathbf{p}_k\|_2 \\ \arctan(\frac{p_y - y}{p_x - x}) \\ \arccos(p_z - z, r) \end{bmatrix} + \mathbf{v}_i^r \\
&= [r\ \alpha\ \beta]^T + \mathbf{v}_i^r
\end{aligned}
\tag{6.2}
$$

Dynamic Graphs   The segmentation method proposed in chapter 5 relied on the odometry-based positions to segment incoming points and update the patch adjacency graph. Ideally, as new estimates of each point's location become available, it could prove beneficial to update the segmentation based on these latest estimates and modify the factor graph accordingly. This could also lead to smaller (if maybe denser) graphs as a match between two or more adjacent patches could then be replaced by a single patch variable.

Multiple Agents, Sessions

One of the most interesting areas in the marine robotics field is that of multi-agent systems - the recent availability of low-cost platforms [111], sensors, and navigation packages [93] has the potential to finally deliver on the long-promised claims of increased robustness

and reduced execution times that a mission performed by multiple robots can offer. While high-end platforms will continue to be the main workhorse for many applications, multi-agent systems may render certain use cases feasible. One such example is that of ship hull inspection: high-value ships and platforms, such as the latest generations of container ships and aircraft carriers have large submerged surface areas on the order of thousands of square meters[1]. Detail inspection of these areas in reasonable amounts of time would require the deployment of multiple inspection platforms simultaneously, posing interesting research and development problems. In fact, some of the research performed in the scope of ONR's ship hull inspection project (under which this research took place) is headed in this direction.

---

[1]*Mærsk's Triple E* class of container ships, for instance, has length, beam, and draft dimensions of 399, 59, and 16 meters, respectively.

Bibliography

[1]   *3DM-GX-5 Attitude and Heading Reference System (AHRS)*. LORD Sensing. URL: https://www.microstrain.com/sites/default/files/3dm-gx5-25_datasheet_8400-0093_2.pdf.

[2]   D. A. Abraham, J. M. Gelb, and A. W. Oldag. "Background and Clutter Mixture Distributions for Active Sonar Statistics". In: *IEEE J. Oceanic Eng.* 36.2 (Apr. 2011), pp. 231–247. ISSN: 0364-9059. DOI: 10.1109/JOE.2010.2102150.

[3]   D. A. Abraham and P. K. Willett. "Active Sonar Detection in Shallow Water using the Page test". In: *IEEE J. Oceanic Eng.* 27.1 (Jan. 2002), pp. 35–46. ISSN: 0364-9059. DOI: 10.1109/48.989883.

[4]   *Absolute Pressure Transducers - Series 3000 & 4000*. Paroscientific, Inc. URL: http://paroscientific.com/pdf/D25_Series_2000_3000_4000.pdf.

[5]   Yong Bai and Qiang Bai. Ed. by Y. Bai and Q. Bai. Boston: Gulf Professional Publishing, 2010. ISBN: 978-1-85617-689-7. DOI: 10.1016/B978-1-85617-689-7.10030-5.

[6]   E. Belcher, W. Hanot, and J. Burch. "Dual-Frequency Identification Sonar (DIDSON)". In: *Proceedings of the 2002 International Symposium on Underwater Technology*. 2002, pp. 187–192. DOI: 10.1109/UT.2002.1002424.

[7]   P. J. Besl and N. D. McKay. "Method for registration of 3-D shapes". In: *Proc. SPIE 1611, Sensor Fusion IV: Control Paradigms and Data Structures*. Vol. 1611. Apr. 1992. DOI: 10.1117/12.57955.

[8]   J.-L. Blanco. *A tutorial on SE(3) Transformation parameterizations and on-manifold optimization*. Tech. rep. 012010. Universidad de Malaga, Aug. 2012.

[9]   M. Bosse et al. "An Atlas framework for scalable mapping". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 2. IEEE, 2003, pp. 1899–1906. DOI: 10.1109/ROBOT.2003.1241872.

[10] M. Bosse et al. "Simultaneous Localization and Map Building in Large-Scale Cyclic Environments Using the Atlas Framework". In: *Int. J. Robotics Research* 23.12 (2004), pp. 1113–1139. DOI: 10.1177/0278364904049393.

[11] N. A. Brokloff. "Matrix algorithm for Doppler sonar navigation". In: *OCEANS*. Vol. 3. Sept. 1994, III/378–III/383 vol.3. DOI: 10.1109/OCEANS.1994.364228.

[12] W. S. Burdic. *Underwater Acoustic System Analysis*. Ed. by A. V. Oppenheim. Prentice-Hall, 1984. ISBN: 0139367160.

[13] A. Burguera, G. Oliver, and Y. González. "Range extraction from underwater Imaging Sonar data". In: *IEEE Conf. Emerging Technologies and Factory Automation (EFTA)*. Sept. 2010, pp. 1–4. DOI: 10.1109/ETFA.2010.5641076.

[14] B. R. Calder. "Robust Automatic Multi-beam Bathymetric Processing". In: Hydrographic Society of America, May 2001. URL: https://scholars.unh.edu/ccom/216.

[15] B. R. Calder and L. A. Mayer. "Automatic processing of high-rate, high-density multibeam echosounder data". In: *Geochemistry, Geophysics, Geosystems* 4.6 (June 2003). ISSN: 1525-2027. DOI: 10.1029/2002GC000486.

[16] D. Cormell. "A new tool in the subsea industry: the Autonomous Inspection Vehicle AIV". In: *The APPEA Journal* 52.2 (2012), p. 659. DOI: 10.1071/AJ11073.

[17] A. Davis and A. Lugsdin. "High speed underwater inspection for port and harbour security using Coda Echoscope 3D sonar". In: *MTS/IEEE OCEANS*. Sept. 2005, 2006–2011 Vol. 3. DOI: 10.1109/OCEANS.2005.1640053.

[18] F. Dellaert and M. Kaess. "Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing". In: *Int. J. Robotics Research* 25.12 (2006), pp. 1181–1203. DOI: 10.1177/0278364906072768.

[19] *DIDSON 300 Standard Version Specifications*. Sound Metrics Corp. Bellevue, WA, USA, Nov. 2012. URL: http://www.soundmetrics.com/Products/DIDSON-Sonars/DIDSON-300m/DIDSON-300-Standard-Version-Specifications.

[20] S. Dugelay and D. A. Abraham. *Reduction of Low Frequency Active Sonar Clutter Through Image Processing*. Tech. rep. SACLANTCEN report SR-272-UU. SACLANT Undersea Research Centre, Dec. 1997.

[21] A. Elfes. "Using occupancy grids for mobile robot perception and navigation". In: *Computer* 22.6 (June 1989), pp. 46–57. ISSN: 0018-9162. DOI: 10.1109/2.30720.

[22] R. M. Eustice, R. Camilli, and H. Singh. "Towards bathymetry-optimized Doppler re-navigation for AUVs". In: *MTS/IEEE OCEANS*. IEEE. 2005, pp. 1430–1436. DOI: 10.1109/OCEANS.2005.1639956.

[23] R. M. Eustice, O. Pizarro, and H. Singh. "Visually augmented navigation in an unstructured environment using a delayed state history". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 1. Apr. 2004, pp. 25–32. DOI: 10.1109/ROBOT.2004.1307124.

[24] R. M. Eustice et al. "Visually Navigating the RMS Titanic with SLAM Information Filters." In: *Robotics: Science and Systems*. Vol. 2005. 2005, pp. 57–64.

[25] N. Fairfield, G. Kantor, and D. Wettergreen. "Real-Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels". In: *J. Field Robotics* 24.1-2 (2007), pp. 03–21. DOI: 10.1002/rob.20165.

[26] N. Fairfield et al. "Autonomous Exploration and Mapping of Flooded Sinkholes". In: *Int. J. Robotics Research* 29.6 (2010), pp. 748–774. DOI: 10.1177/0278364909344779.

[27] J. A. Farrell. *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, 2008. DOI: 10.1036/0071493298.

[28] *Final Report on the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France*. Tech. rep. Bureau d'Enquêtes et d'Analyses pour la securite de l'aviation civile, July 2012. URL: https://www.bea.aero/fileadmin/documents/docspa/2009/f-cp090601.en/pdf/f-cp090601.en.pdf.

[29] N. P. Fofonoff and R. C. Millard Jr. "Algorithms for the computation of fundamental properties of seawater". In: Unesco Technical Papers in Marine Science 44. Unesco, 1983.

[30] C. S. Forbes et al. *Statistical Distributions*. Hoboken, New Jersey: John Wiley & Sons, Nov. 2010. ISBN: 9780470627242. DOI: 10.1002/9780470627242.

[31] D. Fourie. "Multi-Modal and Inertial Sensor Solutions to Navigation-type Factor Graphs". PhD thesis. Cambridge, MA, USA: Massachusetts Inst. of Technology, Sept. 2017. DOI: 10.1575/1912/9305.

[32] *Frequently Asked Questions*. July 2019. URL: https://seabed2030.gebco.net/faq/.

[33] Ø. Hegrenaes and O. Hallingstad. "Model-aided INS with sea current estimation for robust underwater navigation". In: *IEEE J. Oceanic Eng.* 36.2 (2011), pp. 316–337. DOI: 10.1109/JOE.2010.2100470.

[34] *HG1700 Inertial Measurement Unit*. N61-1619-000-001 I 06/1. Honeywell Aerospace. 2018. URL: https://aerospace.honeywell.com/en/products/navigation-and-sensors/hg1700.

[35]   *High Precision Digital Pressure Transmitter - Preciseline*. KELLER America, Inc. Mar. 2019. URL: https://www.kelleramerica.com/pdf-library/High%5C%20Precision% 5C%20Digital%5C%20Pressure%5C%20Transmitter%5C%20Preciseline.pdf.

[36]   A. Hornung et al. "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees". In: *Autonomous Robots* (2013). Software available at http:// octomap.github.com. DOI: 10.1007/s10514-012-9321-0.

[37]   F.S. Hover et al. "A Vehicle System for Autonomous Relative Survey of In-Water Ships". In: *Marine Technology Society Journal* 41.2 (June 2007), pp. 44–55. ISSN: 0025-3324. DOI: 10.4031/002533207787442196.

[38]   E. E. Hundt and E. A. Trautenberg. "Digital Processing of Ultrasonic Data by Deconvolution". In: *IEEE Transactions on Sonics and Ultrasonics* 27.5 (Sept. 1980), pp. 249–252. ISSN: 0018-9537. DOI: 10.1109/T-SU.1980.31185.

[39]   *Inventory of Radioactive Material Resulting from Historical Dumping, Accidents and Losses at Sea*. TECDOC Series 1776. Vienna: International Atomic Energy Agency, 2015. ISBN: 978-92-0-109015-7. URL: https://www.iaea.org/publications/10925/ inventory-of-radioactive-material-resulting-from-historical-dumping-accidents-and-losses-at-sea.

[40]   *Inventory of Radioactive Waste Disposals at Sea*. TECDOC Series 1105. Vienna: International Atomic Energy Agency, 1999. URL: https://www.iaea.org/publications/ 5786/inventory-of-radioactive-waste-disposals-at-sea.

[41]   H. Johannsson et al. "Imaging sonar-aided navigation for autonomous underwater harbor surveillance". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Oct. 2010, pp. 4396–4403. DOI: 10.1109/IROS.2010.5650831.

[42]   N. L. Johnson, A. W. Kemp, and S. Kotz. *Univariate Discrete Distributions*. third. Wiley Series in Probability and Statistics. Hoboken, New Jersey: John Wiley & Sons, Jan. 2005. DOI: 10.1002/0471715816.

[43]   M. Kaess. "Simultaneous localization and mapping with infinite planes". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. May 2015, pp. 4605–4611. DOI: 10.1109/ ICRA.2015.7139837.

[44]   M. Kaess, A. Ranganathan, and F. Dellaert. "iSAM: Incremental Smoothing and Mapping". In: *IEEE Trans. Robotics* 24.6 (Dec. 2008), pp. 1365–1378. ISSN: 1552-3098. DOI: 10.1109/TRO.2008.2006706.

[45]   S. M. Kay. *Fundamentals of statistical singal processing: estimation theory*. Ed. by A. V. Oppenheim. Prentice-Hall Signal Processing Series. Upper Saddle River, New Jersey: Prentice-Hall, 1993. ISBN: 0-13-345711-7.

[46] M. Keller et al. "Real-Time 3D Reconstruction in Dynamic Scenes Using Point-Based Fusion". In: *Int. Conf. 3D Vision*. June 2013. DOI: 10.1109/3DV.2013.9.

[47] J. C. Kinsey and L. L. Whitcomb. "Towards in-situ calibration of gyro and doppler navigation sensors for precision underwater vehicle navigation". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 4. 2002, pp. 4016–4023. DOI: 10.1109/robot.2002.1014364.

[48] J. C. Kinsey and L. L. Whitcomb. "Preliminary Field Experience with the DVLNAV Integrated Navigation System for Oceanographic Submersibles". In: *Control Engineering Practice* 12.12 (2004), pp. 1541–1549.

[49] J. C. Kinsey and R. M. Eusticeand L. L. Whitcomb. "A Survey of Underwater Vehicle Navigation: Recent Advances and New Challenges". In: *IFAC Conf. Manoeuvering and Control of Marine Craft*. Vol. 88. 2006.

[50] C. G. Kunz. "Autonomous Underwater Vehicle Navigation and Mapping in Dynamic, Unstructured Environments". PhD thesis. Cambridge, MA, USA: Massachusetts Inst. of Technology, 2012. DOI: 1721.1/71277.

[51] C. G. Kunz and H. Singh. "Map Building Fusing Acoustic and Visual Information using Autonomous Underwater Vehicles". In: *J. Field Robotics* 30.5 (2013), pp. 763–783. ISSN: 1556-4967. DOI: 10.1002/rob.21473.

[52] G. Kurz and U. D. Hanebeck. "Dynamic Surface Reconstruction by Recursive Fusion of Depth and Position Measurements". In: *Journal of Advances in Information Fusion* 9.1 (2014), pp. 13–26.

[53] R. Laterveer. *Single ping clutter reduction: segmentation using Markov random fields*. Tech. rep. SACLANT Undersea Research Centre, Mar. 1999.

[54] F. Lu and E. Milios. "Globally Consistent Range Scan Alignment for Environment Mapping". In: *Autonomous Robots* 4.4 (1997), pp. 333–349. ISSN: 1573-7527. DOI: 10.1023/A:1008854305733.

[55] X. Lurton. *An Introduction to Underwater Acoustics: Priciples and Applications*. Springer-Praxis, 2002. ISBN: 3540429670.

[56] I. Mahon and S. B. Williams. "SLAM using natural features in an underwater environment". In: *Int. Conf. Control, Automation, Robotics and Vision Conference (ICARCV)*. Vol. 3. Dec. 2004, pp. 2076–2081. DOI: 10.1109/ICARCV.2004.1469484.

[57] C. Mai et al. "Subsea infrastructure inspection: A review study". In: *Int. Conf. on Underwater System Technology (USYS)*. Dec. 2016, pp. 71–76. DOI: 10.1109/USYS.2016.7893928.

[58] A. Mallios et al. "Pose-based SLAM with probabilistic scan matching algorithm using a mechanical scanned imaging sonar". In: *IEEE OCEANS*. May 2009. DOI: 10.1109/OCEANSE.2009.5278219.

[59] A. Mallios et al. "EKF-SLAM for AUV navigation under probabilistic sonar scan-matching". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Oct. 2010, pp. 4404–4411. DOI: 10.1109/IROS.2010.5649246.

[60] A. Mallios et al. "Probabilistic sonar scan matching SLAM for underwater environment". In: *IEEE OCEANS*. May 2010. DOI: 10.1109/OCEANSSYD.2010.5603650.

[61] W. McVicker et al. "Mapping and visualizing ancient water storage systems with an ROV; An approach based on fusing stationary scans within a particle filter". In: *IEEE Int. Conf. Robotics and Biomimetics (ROBIO)*. Dec. 2012, pp. 538–544. DOI: 10.1109/ROBIO.2012.6491022.

[62] *miniSVS Sound Velocity Sensor*. Valeport. Apr. 2016. URL: http://www.valeport.co.uk/Portals/0/Docs/Datasheets/Valeport-miniSVS.pdf.

[63] N. C. Mitchell. "Processing and analysis of Simrad multibeam sonar data". In: *Marine Geophysical Researches* 18.6 (Dec. 1996), pp. 729–739. ISSN: 1573-0581. DOI: 10.1007/BF00313883.

[64] V. Murino and A. Trucco. "Three-dimensional image generation and processing in underwater acoustic vision". In: *Proc. of the IEEE* 88.12 (Dec. 2000), pp. 1903–1948. ISSN: 0018-9219. DOI: 10.1109/5.899059.

[65] V. Murino et al. "3D mosaicing for environment reconstruction". In: *Int. Conf. Pattern Recognition (ICPR)*. Vol. 3. Sept. 2000, pp. 358–362. DOI: 10.1109/ICPR.2000.903559.

[66] F. Nardi, B. Della Corte, and G. Grisetti. "Unified representation and registration of heterogeneous sets of geometric primitives". In: *IEEE Robotics and Automation Letters* 4.2 (Apr. 2019), pp. 625–632. DOI: 10.1109/lra.2019.2891989.

[67] K. Ni, D. Steedly, and F. Dellaert. "Tectonic SAM: Exact, Out-of-Core, Submap-Based SLAM". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Apr. 2007, pp. 1678–1685. DOI: 10.1109/ROBOT.2007.363564.

[68] *Nomenclature for Treating the Motion of a Submerged Body Through a Fluid*. Tech. rep. Technical and Research Bulletin No. 1-5. The Society of Naval Architects and Marine Engineers, Apr. 1950, pp. 1–15.

[69] A. V. Oppenheim and G. C. Verghese. *Signals, Systems and Inference*. Pearson, 2016. ISBN: 978-0133943283.

[70] P. Ozog and R. M. Eustice. "Real-time SLAM with piecewise-planar surface models and sparse 3D point clouds". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Nov. 2013, pp. 1042–1049. DOI: 10.1109/IROS.2013.6696479.

[71] P. Ozog and R. M. Eustice. "Toward long-term, automated ship hull inspection with visual SLAM, explicit surface optimization, and generic graph-sparsification". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. June 2014, pp. 3832–3839. DOI: 10.1109/ICRA.2014.6907415.

[72] P. Ozog, M. Johnson-Roberson, and R. M. Eustice. "Mapping underwater ship hulls using a model-assisted bundle adjustment framework". In: *Robotics and Autonomous Systems* 87 (2017). Special Issue on Localization and Mapping in Challenging Environments, pp. 329–347.

[73] P. Ozog et al. "Building 3D mosaics from an Autonomous Underwater Vehicle, Doppler velocity log, and 2D imaging sonar". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. May 2015, pp. 1137–1143. DOI: 10.1109/ICRA.2015.7139334.

[74] P. Ozog et al. "Long-term mapping techniques for ship hull inspection and surveillance using an autonomous underwater vehicle". In: *J. Field Robotics* 33.3 (2016). Special Issue on Safety, Security and Rescue Robotics, pp. 265–289.

[75] A. Palomer, P. Ridao, and D. Ribas. "Multibeam 3D Underwater SLAM with Probabilistic Registration". In: *Sensors* 16.4 (Apr. 2016), p. 560. DOI: 10.3390/s16040560.

[76] J. Papon et al. "Voxel Cloud Connectivity Segmentation - Supervoxels for Point Clouds". In: *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. Portland, Oregon, June 2013. DOI: 10.1109/cvpr.2013.264.

[77] L. Paull et al. "AUV Navigation and Localization: A Review". In: *IEEE J. Oceanic Eng.* 39.1 (Jan. 2014), pp. 131–149. ISSN: 0364-9059. DOI: 10.1109/JOE.2013.2278891.

[78] *Phins Subsea*. iXblue. Feb. 2019. URL: https://www.ixblue.com/sites/default/files/2019-02/PhinsSubsea_Datasheet.pdf.

[79] K. Picard, B. Brooke, and M. F. Coffin. "Geological insights from Malaysia Airlines flight MH370". In: *EOS* 98 (Mar. 2017). DOI: 10.1029/2017EO069015.

[80] D. Ribas et al. "SLAM using an Imaging Sonar for Partially Structured Underwater Environments". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Oct. 2006, pp. 5040–5045. DOI: 10.1109/IROS.2006.282532.

[81] D. Ribas et al. "Underwater SLAM in man-made structured environments". In: *J. Field Robotics* 25.11-12 (2008), pp. 898–921. DOI: 10.1002/rob.20249.

[82] C. Roman. "Self Consistent Bathymetric Mapping from Robotic Vehicles in the Deep Ocean". PhD thesis. Cambridge, MA, USA: Massachusetts Inst. of Technology, 2005. DOI: 1721.1/39184.

[83] C. Roman and H. Singh. "Micro-bathymetric mapping using acoustic range images". In: *MTS/IEEE OCEANS*. Vol. 3. Nov. 2004, 1574–1579 Vol.3. DOI: 10.1109/OCEANS.2004.1406357.

[84] C. Roman and H. Singh. "Improved vehicle based multibeam bathymetry using sub-maps and SLAM". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Aug. 2005, pp. 3662–3669. DOI: 10.1109/IROS.2005.1545340.

[85] C. Roman and H. Singh. "A Self-Consistent Bathymetric Mapping Algorithm". In: *J. Field Robotics* 24.1-2 (Feb. 2007), pp. 23–50. DOI: 10.1002/rob.20164.

[86] D. Roy. "Discrete Rayleigh distribution". In: *IEEE Transactions on Reliability* 53.2 (June 2004), pp. 255–260. ISSN: 0018-9529. DOI: 10.1109/TR.2004.829161.

[87] M. Ruhnke et al. "Highly accurate maximum likelihood laser mapping by jointly optimizing laser points and robot poses". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. May 2011, pp. 2812–2817. DOI: 10.1109/ICRA.2011.5980220.

[88] M. Ruhnke et al. "Range sensor based model construction by sparse surface adjustment". In: *Advanced Robotics and its Social Impacts*. Oct. 2011, pp. 46–49. DOI: 10.1109/ARSO.2011.6301981.

[89] M. Ruhnke et al. "3D Environment Modeling Based on Surface Primitives". In: *Towards Service Robots for Everyday Environments: Recent Advances in Designing Service Robots for Complex Tasks in Everyday Environments*. Ed. by E. Prassler et al. Berlin, Heidelberg, 2012, pp. 281–300. ISBN: 978-3-642-25116-0. DOI: 10.1007/978-3-642-25116-0_19.

[90] S. Rusinkiewicz. and M. Levoy. "Efficient Variants of the ICP Algorithm." In: *Int. Conf. on 3D Digital Imaging and Modeling (3DIM)*. June 2001, pp. 145–152.

[91] R. B. Rusu and S. Cousins. "3D is here: Point Cloud Library (PCL)". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Shanghai, China, May 2011. DOI: 10.1109/icra.2011.5980567.

[92] N. R. Rypkema. "A Straightforward Derivation of the Matched Filter". Apr. 2019.

[93] N. R. Rypkema. "Underwater and Out of Sight: Towards Ubiquity in Underwater Robotics". PhD thesis. Cambridge, MA, USA: Massachusetts Inst. of Technology, Sept. 2019.

[94]   IHO S-44. *IHO Standards for Hydrographic Surveys.* Special Publication 44. International Hydrographic Organization, Feb. 2008.

[95]   A. Segal, D. Haehnel, and S. Thrun. "Generalized-ICP". In: *Robotics: Science and Systems.* Vol. 2. 4. 2009, p. 435.

[96]   R. C. Smith and P. Cheeseman. "On the Representation and Estimation of Spatial Uncertainty". In: *Int. J. Robotics Research* 5.4 (1986), pp. 56–68. DOI: 10.1177/027836498600500404.

[97]   *SS Richard Montgomery Survey Report 2017.* Tech. rep. Maritime and Coastguard Agency, Jan. 2019. URL: https://www.gov.uk/government/publications/the-ss-richard-montgomery-information-and-survey-reports.

[98]   T. K. Stanton. "Sonar estimates of seafloor microroughness". In: *J. Acoustical Society of America* 75.3 (1984), pp. 809–818. DOI: 10.1121/1.390590.

[99]   T. K. Stanton and C. Clay. "Sonar echo statistics as a remote-sensing tool: Volume and seafloor". In: *IEEE J. Oceanic Eng.* 11.1 (Jan. 1986), pp. 79–96. ISSN: 0364-9059. DOI: 10.1109/JOE.1986.1145139.

[100] W. K. Stewart. "A non-deterministic approach to 3-D modeling underwater". In: *Int. Symp. Unmanned Untethered Submersible Technology (UUST).* Vol. 5. Institute of Electrical and Electronics Engineers (IEEE), 1987, pp. 283–309. DOI: 10.1109/uust.1987.1158599.

[101] W. K. Stewart. *Multisensor Modeling Underwater with Uncertain Information.* Tech. rep. 1143. MIT Artificial Intelligence Laboratory, July 1988. DOI: 1721.1/6980.

[102] L. D. Stone et al. *Search Analysis for the location of the AF447 underwater wreckage.* Tech. rep. Report to Bureau d'Enquêtes et d'Analyses. METRON Scientific Solutions, Jan. 2011.

[103] D. Stutz. "Superpixel Segmentation: An Evaluation". In: *Lecture Notes in Computer Science.* Springer International Publishing, 2015, pp. 555–562. ISBN: 978-3-319-24947-6. DOI: 10.1007/978-3-319-24947-6_46.

[104] P. V. Teixeira et al. "Underwater inspection using sonar-based volumetric submaps". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS).* IEEE, Oct. 2016, pp. 4288–4295. DOI: 10.1109/iros.2016.7759631.

[105] P. V. Teixeira et al. "Multibeam Data Processing for Underwater Mapping". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS).* IEEE, Oct. 2018, pp. 1877–1884. DOI: 10.1109/iros.2018.8594128.

[106] P. V. Teixeira et al. "Dense, Sonar-based Reconstruction of Underwater Scenes". In: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. IEEE, Nov. 2019.

[107] *The Operational Search for MH370*. Tech. rep. Australian Transportation Safety Bureau, Oct. 2017. URL: https://www.atsb.gov.au/media/5773565/operational-search-for-mh370_final_3oct2017.pdf.

[108] S. Thrun. "Learning Occupancy Grid Maps with Forward Sensor Models". In: *Autonomous Robots* 15.2 (Sept. 2003), pp. 111–127. ISSN: 1573-7527. DOI: 10.1023/A:1025584807625.

[109] A. J. B. Trevor, J. G. Rogers, and H. I. Christensen. "Planar surface SLAM with 3D and 2D sensors". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. May 2012, pp. 3041–3048. DOI: 10.1109/ICRA.2012.6225287.

[110] A. J. B. Trevor et al. "Efficient organized point cloud segmentation with connected components". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Workshop on Semantic Perception Mapping and Exploration (SPME). 2013.

[111] A. Underwood and C. Murphy. "Design of a micro-AUV for autonomy development and multi-vehicle systems". In: *IEEE OCEANS*. June 2017. DOI: 10.1109/OCEANSE.2017.8084807.

[112] J. P. Underwood et al. "Error modeling and calibration of exteroceptive sensors for accurate mapping applications". In: *J. Field Robotics* 27.1 (2010), pp. 2–20. ISSN: 1556-4967. DOI: 10.1002/rob.20315.

[113] R. J. Urick. *Principles of Underwater Sound*. Third. McGraw-Hill, 1983. ISBN: 0070660875.

[114] J. Vaganay et al. "Ship Hull Inspection with the HAUV: US Navy and NATO Demonstrations Results". In: *IEEE OCEANS*. Sept. 2006, pp. 1–6. DOI: 10.1109/OCEANS.2006.307039.

[115] M. A. VanMiddlesworth. "Toward Autonomous Underwater Mapping in Partially Structured 3D Environments". MA thesis. Cambridge, MA, USA: Massachusetts Inst. of Technology, Jan. 2014. DOI: 1721.1/87791.

[116] M. A. VanMiddlesworth et al. "Mapping 3D Underwater Environments with Smoothed Submaps". In: *Int. Conf. Field and Service Robotics (FSR)*. Springer International Publishing, 2015, pp. 17–30. ISBN: 978-3-319-07488-7. DOI: 10.1007/978-3-319-07488-7_2.

[117] L. A. Wainstein and V. D. Zubakov. *Extraction of Signals from Noise*. Dover Publications, 1970. ISBN: 0486626253.

[118] A. D. Waite. *Sonar for Practising Engineers*. third. Wiley, Nov. 2001. ISBN: 978-0471497509.

[119] T. Weise et al. "In-hand scanning with online loop closure". In: *IEEE Int. Conf. Computer Vision (ICCV)*. Sept. 2009, pp. 1630–1637. DOI: 10.1109/ICCVW.2009.5457479.

[120] E. Westman and M. Kaess. *Underwater AprilTag SLAM and Calibration for High Precision Robot Localization*. Tech. rep. CMU-RI-TR-18-43. Robotics Institute, Carnegie Mellon University, Oct. 2018.

[121] L. Whitcomb, D. Yoerger, and H. Singh. "Advances in Doppler-based navigation of underwater robotic vehicles". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 1. 1999, pp. 399–406. DOI: 10.1109/robot.1999.770011.

[122] S. B. Williams and I. Mahon. "Simultaneous localisation and mapping on the Great Barrier Reef". In: *IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 2. Apr. 2004, 1771–1776 Vol.2. DOI: 10.1109/ROBOT.2004.1308080.

[123] P. M. Woodward. *Probability and Information Theory, with Applications to Radar*. Ed. by D. W. Fry. Electronics and Waves. Pergamon Press, 1953.

[124] *Workhorse Navigator Doppler Velocity Log*. Teledyne RD Instruments. July 2013. URL: http://www.teledynemarine.com/Lists/Downloads/navigator_datasheet_lr.pdf.

[125] S. Zhang. *The Search for MH370 Revealed Secrets of the Deep Ocean*. 2017. URL: https://www.theatlantic.com/science/archive/2017/03/mh370-search-ocean/518946.