Universitat de Girona

SONAR SCAN MATCHING FOR SIMULTANEOUS LOCALIZATION AND MAPPING IN CONFINED UNDERWATER ENVIRONMENTS

Angelos Mallios

Dipòsit legal: Gi. 989-2014
http://hdl.handle.net/10803/134727

Universitat de Girona

DOCTORAL THESIS

Sonar Scan Matching for Simultaneous Localization and Mapping in Confined Underwater Environments

Angelos Mallios

2014

Universitat de Girona

DOCTORAL THESIS

Sonar Scan Matching for Simultaneous Localization and Mapping in Confined Underwater Environments

Angelos Mallios

2014

Doctoral Program in Technology

Supervised by: Dr. Pere Ridao and Dr. David Ribas

Work submitted to the University of Girona in fulfillment of the requirements for the degree of Doctor of Philosophy

Certificat de Direcció de Tesi

Dr. Pere Ridao i Dr. David Ribas, del Departament d'Arquitectura i Tecnologia de Computadors de la Universitat de Girona,

Declaro:

Que el treball titulat *Sonar Scan Matching for Simultaneous Localization and Mapping in Confined Underwater Environments*, que presenta Angelos Mallios per a l'obtenció del títol de doctor, ha estat realitzat sota la nostra direcció.
I, perquè així consti i tingui els efectes oportuns, signem aquest document.

Girona, Abril de 2014

Dr. Pere Ridao

Dr. David Ribas

στους γονείς μου
και στην Τάλι

Acknowledgments

Although this thesis is a monograph, many people have helped and influenced me from the beginning of this graduate journey, either directly or indirectly. The endless support from my family, friends, colleagues, and advisors made me feel blessed for being surrounded by such generous and patient people. No words can truly express my gratitude to them.

First, I express my gratitude and love to my parents and my sister for their unconditional support, despite me being so far away for so many years, and visiting less often than I would like. My special gratitude goes to Tali who gave an unexpected meaning to this journey and to my life.

I would like then to thank Rich that convinced me (for a while) that I was not too old for graduate studies and encouraged me to start this adventure. Despite being far and extremely busy, he was always available for discussion when things were unclear or blurry for me. In addition, my involvement with his projects in exotic locations around the world was a great break from the lab (I can finally admit it).

It wouldn't have been possible to complete this thesis without the constant encouragement and advice from Pere, to whom I am grateful. He believed in me (more than I did) and I still cannot realize how he managed to be so patient with me. Big thanks also to David for the countless hours explaining Kalman Filters and SLAM, but also for the fun we had when designing robots. I was very fortune to have the best combination of advisors that I could have imagined.

I would like to thank all of my lab mates for the pleasant work environment and their help whenever I asked for it and in whatever form: Marc, Emili, Narcís, Enric, Andres, Lluís, Sebas, Carles, Arnau, Eduard, Simone, Sharad,

Enrique, Albert, Guillem, Juan David, Josep, and the ones from the other side of the hill: Nuno, Rafa, Xevi, Jordi, Tudor, Ricard[2], Miki, Aulinas, Bladimir, Arman, Joseta, Mireia and Anna[2]. There are so many nice moments shared inside and outside the lab with each one of them, that I could easily fill many pages if I tried try to write them all down. Thank you guys.

I would like to thank my *Freesubnet* project mates: Francesco, Andreas, Mara, Szymon, Panos, Fausto and Mernout. We shared papers, endless discussions and fun in almost every corner of Europe and beyond. My appreciation goes to Yvan, Oscar, Sam and Rich, who from different sides of the globe wrote impressive recommendation letters for my fellowship applications. I am sure that was the main reason they were award to me.

My gratitude also goes to Geraldine who spent time on her weekends proofreading this thesis and made it readable. I cannot imagine how painful was for her eyes. Many thanks also to Eva from the Orcadiving centre who provided the means, and helped me with all my crazy diving requests for data collection without complaining.

I also want to express my thanks to my friends who made Girona feel like home: Manuel and Eva, Nuno and Núria, Imma and Joan, Eva and Enrique, Maria and Marc, Àngels and Xavi, Ben and Eva, Elsa, Sussana, Anna, Maria, Magda, Claudia, and Anka. You have made living here a truly wonderful and fun experience. And I cannot forget the funny and relaxing hours with my *Weekends* teammates: Enrique, David, Eva, Andriu, Xavi, Carles, Manuel, Ricardo, and Vicente. If we had a motto, it would have been: Jumping from planes, landing into *paellas*.

Finally, I would like to thank the Hellenic Centre for Marine Research, the EU Marie Curie Actions and the Universitat de Girona for the financial and administrative support of this work.

List of Publications

Publications derived from this thesis

The work developed in this thesis led to the following publications:

Journal Articles

- **Mallios, A.**, Ridao, P., Ribas, D. and Hernández, E. (2014), *Scan Matching SLAM in Underwater Environments.* Autonomous Robots 36, 181-198

- Petillot, Y., Maurelli, F., Valeyrie, N., **Mallios, A.**, Ridao, P., Aulinas, J. and Salvi, J. (2010), *Acoustic-based techniques for autonomous underwater vehicle localization.* Journal of Engineering for the Maritime Environment, 224, 293-307

Book chapter

- **Mallios, A.**, Ridao, P., Ribas, D. and Zandara, S. Roberts, G. and Sutton, R. (Eds.) (2012), *Further Advances in Unmanned Vehicles Sonar based Simultaneous Localization and Mapping for Autonomous Underwater Vehicles.* The Institution of Engineering and Technology, 149-170

Conferences

- **Mallios, A.**, Ridao, P., Carreras, M. and Hernández, E. (2011), *Navigating and Mapping with the Sparus AUV in a Natural and Unstructured Underwater Environment.* In Proceedings of the MTS/IEEE Oceans Conference, Kona

- **Mallios, A.**, Ridao, P., Garcia, R., Maurelli, F. and Petillot, Y. (2010), *On the closed-form covariance estimation for the pIC algorithm.* 7th IFAC Symposium on Intelligent Autonomous Vehicles, 7

- **Mallios, A.**, Ridao, P., Ribas, D. and Hernández, E. (2010) *Probabilistic Sonar Scan Matching SLAM for Underwater Environment.* In Proceedings of the MTS/IEEE Oceans Conference, Sydney

- **Mallios, A.**, Ridao, P., Ribas, D., Maurelli, F. and Petillot, Y. (2010), *EKF-SLAM for AUV navigation under Probabilistic Sonar Scan-Matching.* IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4404 - 4411

- **Mallios, A.**, Ridao, P., Hernández, E., Ribas, D., Maurelli, F. and Petillot, Y. (2009, *Pose-Based SLAM with Probabilistic Scan Matching Algorithm using a Mechanical Scanned Imaging Sonar.* In Proceedings of the MTS/IEEE Oceans Conference, Bremen

- **Mallios, A.**, Ribas, D. and Ridao, P. (2009), *Localization Advances in the Unstructured Underwater Environment.* Proceedings of the 9th Hellenic Symposium of Oceanography and Fishery, 1, pp. 111 - 116

Abstracts, Posters

- **Mallios, A.**, Ridao, P., Carreras, M. and Hernández, E. (2011), *Navigating and Mapping with the Sparus AUV in a Natural and Unstructured Underwater Environment.* In Proceedings of the MTS/IEEE Oceans Conference, Kona

- Carreras, M., Vidal, E., Candela, C., Ribas, D., **Mallios, A.**, Magí, L., Palomeras, N. and Ridao, P. (2013) *SPARUS II AUV, an open platform for the euRathlon underwater robotics challenge.* euRathlon Workshop on Field Robotics

Awards

- Oceans 2011 MTS/IEEE Kona, Student Poster Program. **First Place Award** for the poster: *Navigating and Mapping with the Sparus AUV in a Natural and Unstructured Underwater Environment.*

Other publications

Parallel work at the time of this thesis led to the following publications:

Journal Articles

- Ribas, D., Palomeras, N., Ridao, P., Carreras, M. and **Mallios, A.** (2012), Girona 500 AUV: From Survey to Intervention. IEEE/ASME Transactions on Mechatronics, 17, 46 -53

- Bingham, B., Foley, B., Singh, H., Camilli, R., Dellaporta, K., Eustice, R., **Mallios, A.**, Mindell, D., Roman, C. and Sakellariou, D. (2010), *Robotic Tools for Deep Water Archaeology: Surveying an Ancient Shipwreck with an Autonomous Underwater Vehicle.* Journal of Field Robotics, 27, 702-714

- Foley, B., Sakellariou, D., Dellaporta, K., Bingham, B., Camilli, R., Eustice, R., Evagelistis, D., Ferrini, V., Hansson, M., Katsaros, K., Kourkoumelis, D., **Mallios, A.**, Micha, P., Mindell, D., Roman, C., Singh, H., Switzer, D. and Theodoulou, T. (2009), *New Methods for Underwater Archaeology: The 2005 Chios Ancient Shipwreck Survey.* HESPERIA - Journal of the American Archaeological School at Athens, 78, 269-305

Conferences

- Gracias, N., Ridao, P., Garcia, R., Escartín, J., L'Hour, M., Cibecchini, F., Campos, R., Carreras, M., Ribas, D., Palomeras, N., Magi, L., Palomer, A., Nicosevici, T., Prados, R., Hegedüs, R., Neumann, L., de Filippo, F. and **Mallios, A.** (2013), *Mapping the Moon: Using a lightweight AUV to survey the site of the 17th Century ship 'La Lune'.* In Proceedings of the MTS/IEEE Oceans Conference, Bergen

- Palomer, A., Ridao, P., Ribas, D., **Mallios, A.**, Gracias, N. and Vallicrosa, G. (2013) *Bathymetry-based SLAM with Difference of Normals Point-Cloud subsampling and probabilistic ICP registration.* In Proceedings of the MTS/IEEE Oceans Conference, Bergen

- Zandara, S., Ridao, P., Ribas, D., **Mallios, A.** and Palomer, A. (2013), *Probabilistic Surface Matching for Bathymetry Based SLAM.* IEEE International Conference on Robotics and Automation (ICRA), pp. 40-45

- Ribas, D., Ridao, P., **Mallios, A.** and Palomeras, N. (2012), *Delayed State Information Filter for Aided-USBL AUV Navigation.* IEEE International Conference on Robotics and Automation (ICRA), 4898 -4903

- Zandara, S., Ridao, P., Ribas, D., Campos, R. and **Mallios, A.** (2011), *Kornati Bathymetry Survey Data-Set for Navigation and Mapping.* 19th Mediterranean Conference on Control Automation (MED), 443-448

- Maurelli, F., **Mallios, A.**, Krupinski, S., Petillot, Y. and Ridao, P. (2007), *Speeding-up Particle Convergence with Probabilistic Active Localisation for AUV.* 7th IFAC Symposium on Intelligent Autonomous Vehicles, Lecce, Italy, 7

- Hernández, E., Ridao, P., Ribas, D. and **Mallios, A.** (2009), *Probabilistic sonar scan matching for an AUV.* IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 255-260

- Hernández, E., Ridao, P., Ribas, D., **Mallios, A.** and Carreras, M. (2009), *Occupancy Grid Mapping in an Underwater Structured Environment.* 8th Conference on Maneuvering and Control of Marine Craft , (MCMC2009)

- Krupinski, S., Maurelli, F., **Mallios, A.**, Sotiropoulos, P. and Palmer, T. (2009), *Towards AUV docking on sub-sea structures.* In Proceedings of the MTS/IEEE Oceans Conference, Bremen

- Maurelli, F., **Mallios, A.**, Ribas, D., Ridao, P. and Petillot, Y. (2009), *Particle Filter Based AUV Localization using Imaging Sonar.* 8th Conference on Maneuvering and Control of Marine Craft , (MCMC2009)

- Maurelli, F., Petillot, Y., **Mallios, A.**, Krupinski, S., Haraksim, R. and Sotiropoulos, P. (2009), *Investigation of portability of space docking techniques for autonomous underwater docking.* In Proceedings of the MTS/IEEE Oceans Conference, Bremen

- Maurelli, F., Petillot, Y., **Mallios, A.**, Ridao, P. and Krupinski, S. (2009), *Sonar-based AUV localization using an improved particle filter approach.* In Proceedings of the MTS/IEEE Oceans Conference, Bremen

- Palmer, T., Ribas, D., Ridao, P. and **Mallios, A.** (2009), *Vision based localization system for AUV docking on subsea intervention panels.* In Proceedings of the MTS/IEEE Oceans Conference, Bremen

Abstracts, Posters

- Nomikou, P., Escartín, J., Ridao, P., Sakellariou, D., Camilli, R., Ballu, V., Moreira, M., Mevel, C., **Mallios, A.**, Deplus, C., Andreani, M., Pot, O., Garcia, R., Rouzie, L., Gabsi, T., Campos, R., Gracias, N., Hurtos, N., Magi, L., Palomeras, N. and Ribas, D. (2012), *Preliminary submarine monitoring of Santorini Caldera: Hydrothermal activity, and seafloor deformation.* Volcanism of the Southern Aegean in the frame of the broader Mediterranean area (VOLSAM)

- Sakellariou, D., Rousakis, G., Maroulakis, S., P. Georgiou, S. K., Henderson, J., Gallou, C., Spondylis, I., Pizarro, O., **Mallios, A.**, Hogarth, P. and Flemming, N. (2011), *The Late Neolithic – Early Helladic Pavlopetri submerged city: innovative mapping techniques and reconstruction of the prehistoric landscape.* DEGUWA Abstracts: In Poseidons Realm XVI: "Early Seafaring of the Mediterranean Sea - From the Beginning until the late Bronze Age 800 BC"

- Pizarro, O., Friedman, A., Johnson-Roberson, M., Williams, S., Mahon, I., Camilli, R., Camilli, L. and **Mallios, A.** (2010), *High-resolution optical and acoustic 3D seafloor reconstructions from robot and diver-based surveys.* XIX Congress of the Carpathian-Balkan geological association, pp. 314

Workshops

- Carreras, M., Candela, C., Ribas, D., **Mallios, A.**, Magí, L., Vidal, E., Palomeras, N. and Ridao, P. (2013), *SPARUS II, design of a lightweight hovering AUV.* MARTECH, 5th International Workshop on Marine Technology, pp. 58-59

- Ridao, P., Ribas, D., Palomeras, N., Carreras, M., **Mallios, A.**, Hurtós, N., Gracias, N., Magí, L., Garcia, R., Campos, R., Prados, R. and Escartín, J. (2013), *Operational validation of GIRONA500 AUV.* MARTECH, 5th International Workshop on Marine Technology, pp. 61

- Hernández, E., Carreras, M., Ridao, P. and **Mallios, A**. (2012), Homotopic Path Planning for an AUV on Maps Improved with Scan Matching. IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV)

- Zandara, S., Ridao, P., **Mallios, A**. and Ribas, D. (2012), *MBpIC-SLAM: Probabilistic Surface Matching for Bathymetry Based SLAM.* 3rd

IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles, Porto, Portugal

Acronyms and Abbreviations

2D	two-dimensional	20
3D	three-dimensional	16
AHRS	attitude and heading reference system	26
ASEKF	augmented state EKF	7
AUV	autonomous underwater vehicle	2
BA	bundle adjustment	56
DR	dead-reckoning	7
DGPS	differential global positioning system	17
DOF	degrees of freedom	26
DVL	Doppler velocity log	7
EIF	extended infomation filter	13
EKF	extended Kalman filter	13
ESEIF	exactly sparse extended information filter	18
FLS	forward looking sonar	18
FMT	Fourier-Mellin transform	44
GPS	global positioning system	10
GRV	Gaussian random variable	31
I-AUV	intervention autonomous underwater vehicle	2
ICNN	individual compatibility nearest neighbor	52
ICP	iterative closest points	20

IDC iterative dual correspondence 39

INS inertial navigation system 10

LBL long base line 10

LF/SoG likelihood field with sum of Gaussians 46

LS least squares 19

MbICP metric-based iterative closest point 39

MSIS mechanically scanned imaging sonar 7

MSISpIC mechanically scanned imaging sonar probabilistic iterative correspondence 41

MRU motion reference unit 7

NDT normal distributions transform 46

NLLS non-linear least squares

PF particle filter 13

pIC probabilistic iterative correspondence 40

ROV remotely operated vehicle 2

SBL short base line 11

SDNC splats distance normalized cut 89

SLAM simultaneous localization and mapping 6

spIC sonar probabilistic iterative correspondence 40

UdG Universitat de Girona 71

USBL ultra short baseline 11

UUV unmanned underwater vehicle 2

List of Figures

List of Tables

List of Algorithms

Contents

Synopsis

This thesis presents the development of a localization and mapping algorithm for an autonomous underwater vehicle (AUV). It is based on probabilistic scan matching of raw sonar scans within a pose-based simultaneous localization and mapping (SLAM) framework.

To address the motion-induced distortions affecting the generation of full sector scans, an extended Kalman filter (EKF) is used to estimate the robot motion during that scan. The filter uses a constant velocity model with acceleration noise for motion prediction. Velocities from Doppler velocity log (DVL) and heading measurements from attitude and heading reference system (AHRS) are fed asynchronously and update the state. The scan is undistorted by compounding the relative robot position in the scan, with the range and bearing measurements of the beams gathered by the sonar. Assuming Gaussian noise, the algorithm is able to estimate the uncertainty of the sonar measurements with respect to a frame located at the center of the scan.

For estimating the global trajectory of the vehicle, a second filter, an augmented state EKF (ASEKF), stores the pose of the vehicle where each full scan was completed. Each new full scan is cross registered with all the previous scans that are in a certain range and a modified probabilistic iterative correspondence (pIC) algorithm is applied. This technique has a twofold effect: first, we obtain a better estimate of the vehicle's displacement that is then used to update the ASEKF, and second, loop-closing events are updated automatically and simultaneously. In addition, we present a closed form method for estimating the uncertainty of the scan matching result.

The proposed method is well suited for confined environments including

but not limited to: geological folds, boulder areas, caves, man-made walls and structures, where a horizontal scanning sonar can constantly detect and distinguish its surroundings over most of the vehicle's trajectory.

The method was tested with three real-world datasets: one obtained in an abandoned marina during an engineering test mission, and two additional ones in the natural environment of underwater cavern systems. In the marina dataset, the results show the quality of our algorithm by comparing it to the ground truth from a global positioning system (GPS) receiver and to other previously published algorithms. For the cavern datasets, the results are compared against fixed ground truth points that the vehicle visits twice along the trajectory it travels. In all the experiments the trajectory correction is notable and the unoptimized algorithm execution time is much faster than the experiment time, indicating the potential of the algorithm for real-time on-board AUV operation.

Resum

Aquesta tesis presenta el desenvolupament d'un algoritme de localització i mapeig per un vehicle submarí autònom (AUV). L'algoritme es basa en tècniques probabilístiques de *scan matching* utilitzant *scans* de sonar dins un sistema de localització i construcció simultània de mapes (SLAM) basat en posició.

S'utilitza un filtre de Kalman extès (EKF) per estimar el moviment del robot durant l'adquisició de l'*scan* i així solucionar les distorsions causades pel moviment al llarg de l'adquisició d'un *scan* complet. Aquest filtre utilitza un model de velocitat constant amb l'acceleració com a soroll per predir el moviment. Velocitats provinents d'un *Doppler velocity log* (DVL) i mesures d'una unitat d'orientació s'introdueixen al filtre asíncronament per tal d'actualitzar l'estat. La distorsió de l'*scan* s'elimina gràcies a la composició de la posició relativa del robot dins l'*scan* amb les mesures de rang i la orientació dels feixos del sónar. Assumint un soroll Gaussià, l'algoritme és capaç d'estimar la incertesa de les mesures del sónar respecte a un sistema de coordenades situat al centre de l'*scan*.

Per estimar la trajectòria global del vehicle, un segon filtre, en aquest cas un EKF d'estat augmentat (ASEKF), guarda la posició del vehicle on es finalitza l'adquisició de cada *scan* complet. Cada nou *scan* es registra amb tots els *scans* previs localitzats dins un rang determinat i s'aplica una modificació de l'algoritme *probabilistic iterative correspondence* (pIC). Aquesta tècnica té dos propòsits: primer, obtenir una millor estimació del desplaçament del vehicle que posteriorment s'utilitzarà per actualitzar l'estat del ASEKF, i segon, actualitzar automàticament i simultàniament events de tancament de bucles. A més a més, es presenta un mètode de formulació tancada per

estimar la incertesa del resultat de l'*scan matching*. El mètode proposat és adequat per entorns confinats, com per exemple: plecs geològics, zones rocoses, coves, parets artificials i estructures on un sonar d'escaneig horitzontal pugui constantment detectar i distingir el seu voltant al llarg de la major part de la trajectòria del vehicle.

El mètode s'ha provat en tres *datasets* adquirits en entorns reals: un obtingut en una marina abonadonada i dos més en l'entorn natural d'un sistema de coves submarí. Els resultats presentats pel *dataset* de la marina demostren la qualitat de l'algoritme comparant-lo amb mesures d'un sistema de posicionament global (GPS) i amb dos algoritmes prèviament publicats. Pel què fa els *datasets* de la cova, els resultats es comparen amb punts de referència coneguts i fixes que el vehicle revisita al llarg de la trajectòria. En tots els experiments la correcció de la trajectòria és notable i el temps d'execució de l'algoritme és molt més ràpid que el temps de l'experiment, indicant així el potencial de l'algoritme per operar en temps real des d'un AUV.

Resumen

Esta tesis presenta el desarrollo de un algoritmo de localización y mapeo para un vehiculo submarino autónomo (AUV). El algoritmo se basa en técnicas probabilísticas de *scan matching* utilizando *scans* de sónar dentro un sistema de localización y construcción simultánea de mapas (SLAM) basado en posición.

Se utiliza un filtro de Kalman extendido (EKF) para estimar el movimiento del robot durante la adquisición del scan y así solucionar las distorsiones causadas por al movimiento a lo largo de la adquisición de un *scan* completo. Este filtro utiliza un modelo de velocidad constante con la acceleracíon como ruido para predir el movimiento. Velocidades provinientes de un *Doppler velocity log* (DVL) y medidas de un sistema de orientación se introducen al filtro asíncronamente para actualizar el estado. La distorsión del scan se elimina gracias a la composición de la posición relativa del robot dentro del scan con las medidas de rango y orientación de los haces del sónar. Asumiendo un ruido Gaussiano, el algortimo es capaz de estimar la incertidumbre de las medidas del sónar respecto a un sistema de coordenadas situado en el centro del *scan*.

Para estimar la trayectoria global del vehículo, un segundo filtro, en este caso un EKF de estado augmentado (ASEKF), guarda la posición del vehículo dónde finaliza la adquisición de cada scan completo. Cada nuevo scan se registra con todos los scans previos localizados dentro de un rango determinado y se aplica una modificación del algoritmo *probabilistic iterative correspondence* (pIC). Esta técnica tiene dos propósitos: primero, obtener una mejor estimación del desplazamiento del vehículo que posteriormente es utilizada para actualizar el estado del ASEKF, y segundo, actualizar

automáticamente y simultáneamente eventos de cierre de bucles. Además, se presenta un método de formulación cerrada para estimar la incertidumbre del resultado del *scan matching*. El método propuesto es adecuado para entornos confinados, como por ejemplo: pliegues geológicos, zonas rocosas, cuevas, paredes artificiales y estructuras donde un sónar de escaneo horizontal pueda constantemente detectar y distinguir su alrededor a lo largo de la mayor parte de la trayectoria del vehículo.

El método se ha probado con tres *datasets* adquiridos en entornos reales: uno obtenido en una marina abandonada y dos más en el entorno natural de un sistema de cuevas submarinas. Los resultados presentados para el *dataset* de la marina demuestran la calidad del algoritmo comparandolo con medidas de un sistema de posicionamiento global (GPS) y con dos algoritmos previamente publicados. Respecto a los *datasets* de la cueva, los resultados se comparan con puntos de referencia conocidos y fijos que el vehículo re-visita a lo largo de la trayectoria. En todos los experimentos la corrección de la trayectoria es notable y el tiempo de ejecución del algoritmo es mucho más rápido que el tiempo del experimento, indicando por lo tanto el potencial del algoritmo para operar en tiempo real desde un AUV.

Chapter 1

Introduction

Over the last couple of decades, interest in underwater research has increased, particular for deep water. This is not surprising as the largest percentage of our planet is covered with water, the oceans greatly influence earth's climate system, and with an equivalent percentage of planet's natural resources underwater, oceans have a great social importance. While most of the mainland has been explored and the majority of natural resource deposits have been identified, the oceans hold most of their secrets deep inside. Although deep water is an environment as hostile for humans as are the Moon and Mars, it seems that we know more about the latter than we do about our oceans. The main reason can be traced back to the fundamental properties of the water; it provides a poor medium for the propagation and penetration of electromagnetic frequencies, which significantly constrains remote sensing techniques (Medwin and Clay, 1998).

The advance of underwater vehicle technology in the recent years has open the road to scientists for regularly reach the sea floor down to 6,000 meters, and has enabled the research of approximately 98 % of the world's ocean floor or 70 % of the Earth's surface (Emig and Geistdoerfer, 2004; Gage and Tyler, 1991). However, only 5 % of the underwater domain can be consider explored (NOAA, 2013).

1.1 Autonomous Underwater Vehicle Applications

Recently, unmanned underwater vehicles (UUVs) have become the standard tool for the exploration, inspection and intervention in both oceans and inland waters. The development of remotely operated vehicles (ROVs) during the 1980s had a huge impact in various application areas. In the oceanographic field for instance, ROVs have reduced the need for manned submersibles, allowing for a safer observation of continuous operations from a few hours to over a number of days. Moreover, in the industrial world, these robots play an important role for the inspection and construction of submerged structures (cables, pipes, dams, offshore platforms, etc.), allowing intervention as well as rescue operations. However, nothing comes without a price. The tether cable imposes important limitations such as the need for a large and expensive support ship and restrictions on the work area, which increase the cost and difficulty of the operations.

These limitations triggered research into a new generation of autonomous underwater vehicles (AUVs) that eliminate the physical connection with surface vessels. Most of these vehicles are prototypes developed by research centers and relatively few of them are currently commercially available but is a rapidly growth industry (Douglas-Westwood, 2012). AUVs have been initially developed for collecting data and samples in addition to inspection capabilities. Pushing the technology forward, a new class of vehicles has emerged very recently from leading research institutions: the intervention autonomous underwater vehicle (I-AUV) (Ribas et al., 2012). Equipped with manipulators they embody the idea of a future substitute for the expensive ROV operations in routine manipulation tasks. They are designed to be operated very close to the seabed or in close proximity to industrial structures such as those used by the offshore industry, which means that precise localization is more than crucial for them.

As explained in Davis (1996), several potential AUV applications are being investigated by various organizations around the world: environmental monitoring, oceanographic research, and maintenance/monitoring of under-

water structures are just a few examples. AUVs are attractive for use in these areas because of their size and their non-reliance on continuous human attendance. Moreover, in confined environments such as under ice (Kunz et al., 2009), AUVs perform better than typical ROVs as the danger of the umbilical entanglement is removed. Given the potential applications and advantages of AUVs, it is no wonder that academic, oceanographic institutions and commercial organizations around the world are investing and using them as research platforms.

The development of AUVs has offered numerous advantages, but has also presented new challenges. Free from the burden of an umbilical cable connecting them to the support ship, they are a quantum leap from widely used ROVs providing a stable platform for the new high resolution scientific sensors. Advances in battery systems and in low power sensors have made it possible to extend the endurance of an AUV to several weeks underwater and the distances traveled to the tens of kilometers (Schofield et al., 2007).

1.2 Motivation

Despite the recent advances in AUV navigation techniques, reliably localizing the vehicle in unstructured and unconstrained areas is still a challenging problem. As an example, in most of the cases, AUVs have to survey large areas unattended and to ensure desirable coverage. To achieve the best possible navigation accuracy, a set of transponders are deployed in the nearby area but with the disadvantages of consuming expensive ship time and complicating operations. Without any external infrastructure providing absolute position measurements, an AUV's trajectory is prone to drift over time at a rate of 0.1 - 1 % of traveled distance, depending on the quality of the sensors employed (Kinsey et al., 2006). For this reason, the majority of AUV operations are performed in open water where the seafloor is relatively even and there are limited obstacles or constraints. Open water also provides a safe space for the vehicle's drift.

Nevertheless, in a plethora of potential applications in confined environments that are currently served either by ROVs, divers, or not served at all,

AUVs can provide a cost effective and safe alternative in the near future.

Offshore oil platforms are underwater structured environments where routine inspection and maintenance is performed by ROVs or divers, depending on the depth and complexity. As oil platforms are being constructed in increasingly deep waters, more efficient solutions are needed. Recent research projects have been focusing on this area and demonstrating AUV's potential to automate tasks such as structures inspection or basic intervention of valve handling (Evans et al., 2003; Marani et al., 2009; Carrera et al., 2013). Although safely navigating an AUV in close proximity to such an environment is a challenge, the structures (pipes, colons, etc) can be relatively easily perceived by AUV sensors, which provide valuable information for environmental interpretation. Moreover, the infrastructure itself can be equipped with acoustic and visual features that allow the AUV to identify them and navigate in relation to them.

However, far from man-made structures, in the natural underwater environment autonomous navigation becomes more complicated. Geological formations (e.g., faults, canyon walls, hydrothermal vents) are highly unstructured and do not resemble any geometrical shape that an algorithm can recognize and around which an AUV can safely navigate. For that reason, close proximity surveys in those areas are performed by ROVs and if AUVs are used, they conduct their survey from a relatively long distance that facilitates obstacle avoidance, sacrificing detailed mapping or any intervention capability.

Moving to more complex environments, underwater caves are the ultimate exploration areas. Umbilical entanglement is the main factor that prevents ROVs to enter cave systems and divers remain the dominant explorers for these areas. Portable survey systems allow for accurate mapping (am Ende, 2001), however the limited autonomy of the divers, and the complexity and the size of the caves that can be few kilometers long with multiple pathways, make surveys very difficult to organize and require that they last several weeks over repeated periods (Farr, 1991). Unfortunately, the higher cost is on human life. As cave divers push the boundaries trying to unveil places where no one has ever been before, many have lost their life making cave diving one

of the highest risk activities (Fock, 2013).

Further from the industrial or exploration fields, AUVs can potentially also have a social impact. During or following a shipwreck, an inspection is almost always needed to detect hydrocarbon leaks, block or drain them if possible, and most importantly, to search for and recover victims. In shallow waters, these tasks are performed by divers with a risk similar to cave diving, and in deeper water ROVs are used for inspecting the wreck from outside because the umbilical prohibits any penetration. A small AUV capable of navigating inside the wreck would be able to provide valuable information fast, regardless of the depth, while removing any human risk. The possibility that such an operation can save lives should not be neglected, as was demonstrated recently when a survivor rescued by a diving team from a capsized tug vessel after having spend three days at 30 meters depth (BBC News, 2013).

With the problem of navigating AUVs in open waters near to a general accepted solution, research efforts are being focused on more complex environments such as man-made structures (Ridao et al., 2014) and Arctic regions (Stone et al., 2010), which are being primarily driven by the energy industry. An AUV capable of safe navigation in complex with physical restrictions environments will be an invaluable tool in industry, science, and society.

1.3 Objectives

Navigating an AUV in complex and confined environments combine a handful of robotics research fields. The vehicle should:

- meaningfully **interpret** the data from its on-board **sensors**,
- build the **map** of the area in which it operates,
- be able to **localize** itself on that map,
- be aware and **avoid** of any **obstacles** around or on its way and finally,
- based on all the previous information, **plan** accordingly its **path** to the goal.

The fundamental steps in autonomous navigation are localization and mapping. The objective of this thesis is focusing on:

The development of a localization and mapping technique for an autonomous underwater vehicle that navigates in confined and possibly unstructured environments, using a mechanically scanned imaging sonar as principal perception sensor.

The last couple of decades have shown very promising results from a number of studies in mobile robotics, during which techniques had developed to address the localization problem without external aid. In particular, the so-called simultaneous localization and mapping (SLAM) techniques have been broadly and successfully applied to indoor and outdoor environments (Bailey and Durrant-Whyte, 2006). Underwater SLAM for AUVs can be classified, from the sensors' vantage point, in two main categories: vision and sonar based SLAM. Underwater vision, in the best conditions, is restricted to a few meters from the vehicle but it is normally rich in information and of very good resolution. On the other hand, sonar has the ability of long distance penetration, even in turbulent waters, but with the downside of sparse and noisy readings. In both cases, due to the physical properties of the media, the unstructured and hostile environment, and the poor quality of the data gathered with a conventional sensor suite, underwater SLAM is still very challenging. For this reason, when compared with land robotics, very few underwater SLAM algorithms working with real data have been reported in the literature (see section 2.2).

This thesis is a contribution in that area, proposing a pose-based algorithm to solve the full SLAM problem of an AUV navigating in a confined, unknown and possibly unstructured environment. For the purpose of this work, we define as confined underwater environment the one that has structures (natural or not) where a horizontal scanning sonar can constantly detect and distinguish from the surroundings for the most time of the vehicle's trajectory. This can include but not restricted to: geological folds, boulders area, inside caves, man-made walls and structures.

The proposed method first estimates the local path traveled by the robot while forming the acoustic image (scan) with range data coming from a mechanically scanned imaging sonar (MSIS), taking into account the robot dead-reckoning (DR) displacements estimated from Doppler velocity log (DVL) and motion reference unit (MRU) sensors. This provides position estimates for correcting the distortions that the vehicle motion produces in the scans. Secondly, consecutive scans are cross-registered using a probabilistic scan matching technique for estimating the displacements of the vehicle, including the uncertainty of the scan matching result. Finally, an augmented state EKF (ASEKF) estimates and keeps the registered scans poses. The raw data from the sensors are processed and fused in-line without any prior structural information or initial pose to be considered.

1.4 Thesis outline

The contents of this thesis are divided into the following chapters.

Chapter 2 presents a survey of current conventional underwater localization techniques and the state-of-the-art on SLAM algorithms with a focus on the underwater domain.

Chapter 3 discusses errors that a moving vehicle introduces to a scan image that is formed by a mechanically scanned imaging sonar. It presents an algorithm that tracks the vehicle position for each individual beam and at the end of a scan sector, reconstructs the corrected scan, referenced in a position of interest.

Chapter 4 provides the categories of scan matching algorithms and the major contributions.

Chapter 5 explains the probabilistic scan matching algorithm that used in this thesis and also presents a closed-form formulation of the scan matching uncertainty.

Chapter 6 presents the main augmented state EKF in a simultaneous localization and mapping framework tailored for the underwater domain.

Chapter 7 describes the autonomous underwater vehicles that were used during the experiments, the datasets and the results of three real world experiments: One in the man-made environment of a marina, and two in the natural environment of underwater cave systems.

Chapter 8 is a discussion about this work and future research areas and opportunities.

Chapter 2

State of the art

Underwater navigation is one of the most significant challenges for AUVs. In other words, it is very difficult to determine the vehicle's position within its environment to enable it to take the correct actions leading to the vehicle successfully accomplishing its mission. The ideal robot should localize itself in the environment using its own sensors without the aid of external infrastructure in much the same way that living beings do.

This thesis focuses on to this problem in the underwater domain and, as mentioned in the objectives (section 1.3), this work combines two main research areas: probabilistic scan matching in a SLAM framework.

This chapter, first briefly presents the conventional underwater localization techniques that are commonly employed in real life operations and surveys, and the main drawbacks behind these techniques: the need for external aid in keeping navigational errors bounded. Then, an overview is following of the most relevant SLAM algorithms that try to address this problem. In addition to publications that introduce fundamental theoretical background, we will focus on publications present field-tested techniques, especially when they have been applied in underwater environments.

2.1 Conventional techniques for underwater localization

Traditionally, the problem of underwater localization has been addressed in various ways, none of which are fully adequate. From the early years of underwater vehicles, depth, altitude, heading, and attitude have been estimated with high bandwidth and accurate internal sensors but horizontal (x, y) positioning lacks such a sensor. To fill the gap, a number of techniques have been proposed over the years and the most conventional techniques for real operations are presented below.

Dead-reckoning based on inertial navigation system (INS) and DVL is a common means of obtaining motion information by measuring accelerations and/or velocities with respect to the seafloor. Position can be estimated by integration but is subject to unbounded growth in error. The positional error growths with time makes their use impractical as standalone sensors for long-term navigation (Kuritsky and Goldstein, 1990). One way to avoid this problem, is for the vehicle to surface from time to time and obtain navigation updates from a global positioning system (GPS) receiver. Glider type AUVs are an example of vehicles utilizing this technique that simplifies their navigation estimation. However, this strategy is not adequate for the majority of AUV applications, such as the ones that require continuous high precision navigation (e.g., bathymetric or photo-mosaic surveys). Moreover, as the application depth increases, the energy efficient decreases with the amount of time the vehicle needs for ascending to acquire GPS signals and descending to continue the survey.

To achieve the best possible navigation accuracy for long term underwater positioning, a common technique is to deploy a set of acoustic transceivers in the area of interest. The transceivers forms a long base line (LBL), and the vehicle's position can be estimated with the accuracy of a few centimeters through acoustic trilateration and fusion of the dead-reckoning information (Bingham et al., 2010). The disadvantage of such a technique is that the deployment, calibration, and recovery of the transponders consumes costly

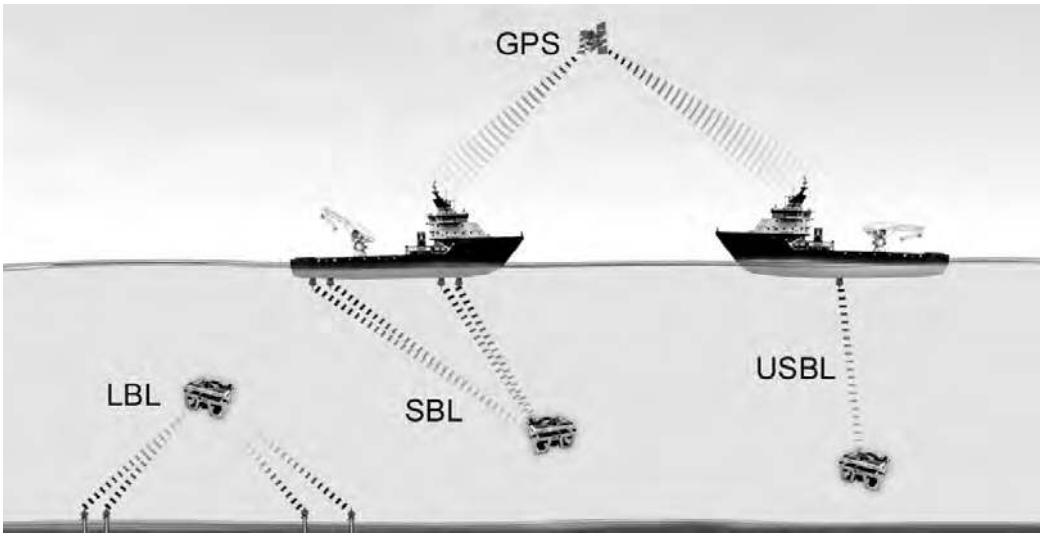


Figure 2.1: Typical configurations for AUV localization using LBL, SBL and USBL (Ribas et al., 2008).

ship time and complicates operations that span a relatively small area of coverage. Alternatively, the acoustic transceivers can be installed on the support ship in a way that forms a short base line (SBL) or an ultra short baseline (USBL) (Fig. 2.1). In this case, the disadvantages are the reduced precision (meter scale) and the fact that the support ship must always be following the vehicle to ensure that it remains within the system's acoustic cone, preventing it from conducting other research during AUV operations (Milne, 1983).

For these reasons, researchers are focusing on other ways to achieve localization of an AUV without the need of external hardware. Map matching techniques use information from on board sensors to provide ground-fixed, feature-relative localization given an *a priori* map of the environment (Nygren, 2005; Fairfield and Wettergreen, 2008; Carreno et al., 2010). Variants of this method have been proposed for gravitational anomaly and magnetic field maps but no operational system has been reported yet (Tyren, 1982; Tuoy et al., 1993; Zhang et al., 2004). Unfortunately, a proper *a priori* map is not usually available for the areas of interest and its production requires extra survey time which can significantly increase the operations budget.

In this section, the conventional operational underwater localization techniques have been briefly presented. For extensive reviews on underwater localization methods, the papers from Stutters et al. (2008) and Kinsey et al. (2006) are highly recommended.

2.2 Simultaneous localization and mapping

A fundamental preconditions for truly autonomous navigation are localization and mapping. With localization, the robot can estimate its position and navigate with respect to an environment; and with mapping, this environment is determined and is the basis for further autonomous behavior such as obstacle avoidance and path planning. Compounding consecutive displacement estimations from scan matching algorithms results in an improved trajectory estimate over than of dead-reckoning, but it still provides no means of bounding the position uncertainty over time. If the robot could recognize areas that it has already visited, estimate the error difference from the last visit, and propagate this estimation through out the trajectory history (loop-closure), the error could be bounded. The technique that allows a robot to estimate its position on a map while simultaneously building that map, and using it to refine its position estimate, is known as simultaneous localization and mapping (SLAM).

However, some drawbacks make it very difficult to implement SLAM in real time with the limited resources of a robot. The first drawbacks are consistency over a long-term operation and the computational cost that increases with each observation. Stored information is increasing rapidly as the robot moves and the required real time computational power can overcome the robot's processing power. Another major challenge is the data association problem or how to determine the correspondences between the observed data and the quantities to be estimated. As a consequence the identification of loop-closure events, which is a fundamental part of SLAM, is a hard problem and prone to errors.

In the last two decades, many researchers have worked towards addressing those problems. Their most notable efforts are presented below.

2.2.1 The SLAM problem

From the theoretical point of view, the SLAM problem was first addressed in a probabilistic framework in the early 1990's with the seminal work from Smith et al. (1990). Since then, a significant amount of research has been conducted and a number of algorithms have been proposed to formulate and solve the problem with notable achievements mostly in land mobile robotics, see Durrant-Whyte and Bailey (2006) and Bailey and Durrant-Whyte (2006) for a comprehensive review.

SLAM relies on filtering algorithms which merge the noisy sensor measurements with information from a kinematic or a dynamic motion model of the system. Filtering algorithms commonly used for SLAM can be separated into Gaussian filters, such as extended Kalman filters (EKFs) or extended infomation filters (EIFs), and non-parametric filters such as particle filters (PFs).

The popular formulation of EKF-SLAM from Smith et al. (1990) has been used in many existing works (Tardós et al., 2002; Leonard and Newman, 2003; Ribas et al., 2008; Piniés et al., 2010). However, its limitations are well known because computational costs increase quadratically with the number of features (Thrun et al., 2005).

Other approaches that allow for constant time updates have been proposed based on the observation that many of the off-diagonal elements in the information matrix of the EKF-SLAM are near zero when properly normalized (Paskin, 2003; Thrun et al., 2004; Eustice et al., 2005; Frese, 2006). Recently, computationally efficient solutions have been proposed including the incremental smoothing and mapping (iSAM2) (Kaess et al., 2012; Ni and Dellaert, 2010), and the tectonic SAM2 (TSAM2) (Ni and Dellaert, 2010).

The SLAM problem has also been formulated for non-Gaussian probability distributions using particle filters (Burgard et al., 1996; Thrun, 2001; Montemerlo et al., 2002; Eliazar and Parr, 2004; Grisetti et al., 2007; Fairfield et al., 2007). This technique uses a finite number of possible positions for the robot to represent the uncertainty distribution. The greater the number of particles, the better the description of the uncertainty. However, this produces

a growth in computational complexity and careful memory management is needed for efficient implementation.

Another method for reducing the computational burden is to split the full map into a series of manageable independent sub-maps. In this approach each map is limited to a fixed number of elements (or map size) and therefore the computational cost of updating each map has an upper bound. In the end or when needed, the sub-maps are joined and the full solution is computed. Notable works using this method include: decoupled stochastic mapping (Leonard and Feder, 2001), local map joining (Tardós et al., 2002), constrained local sub-map filter (Williams et al., 2002), constant time SLAM (Leonard and Newman, 2003) and the ATLAS system (Bosse et al., 2004). Paz et al. (2008) with the divide and conquer SLAM, and Huang et al. (2008) with the sparse local sub-map joining SLAM, are two recent algorithms that provide important reductions in computational cost. Piniés et al. (2010) introduced the CI-Graph where the local maps are conditionally independent. In contrast with the previous algorithms, selective information is shared between sub-maps, which allows for information propagation between updates.

An alternative way to formulate the SLAM problem that has attracted much attention recently is the use of pose-based algorithms. In this family of algorithms, the poses and map are internally represented as a graph where the edges are built from particular constraints between each vertex. Its first implementation, CPE-SLAM (Lu and Milios, 1997a; Gutmann and Konolige, 1999) is a method that makes use of dense sensor data, maintaining a network of local constraints between the robot's positions and producing the map through optimization. Its main advantage is that such a representation scales well with the map area because it generally represents only the local constraints and it gives the capability of working with featureless methods.

Bosse et al. (2004) developed the ATLAS system for large scale environments. Similar to that, Estrada et al. (2005) proposed the hierarchical SLAM. Olson et al. (2006) suggest a gradient descent approach which can efficiently correct even large pose-graphs. Grisetti et al. (2009) extend this idea by applying a tree parametrization to increase the convergence speed. Thrun and Montemerlo (2006) present the GraphSLAM which applies variable elim-

ination techniques to reduce the dimensionality of the optimization problem. Recently, Konolige et al. (2010) presented the sparse pose adjustment (SPA) that constructs the linearized system in an efficient way. A general framework for optimization of non-linear least squares problems and to improve pose-graph was proposed by Kümmerle et al. (2011).

2.2.2 Underwater SLAM

The environment in which a robotic system interacts plays a significant role in its performance. Because of the physical characteristics and the complexity of the marine environment, there has been less research for underwater SLAM than for its mobile robotics counterparts. In contrast to mobile robotics, which in most cases has the advantage of long range visibility and high frequency communications, including GPS signals, the perception of underwater vehicles is limited. From the sensors' vantage point, we can separate underwater SLAM in two main categories: vision and sonar based SLAM. Our proposed SLAM algorithm is based on sonar sensor. Therefor, this section focuses more on sonar than vision-based SLAM.

Vision SLAM

Vision sensors are a common choice among many researchers for a variety of reasons. They are relative inexpensive, they can provide large amounts of high resolution information with fast refresh rates, and well-known techniques from decades of research on computer vision can be applied. On the other hand, vision underwater is limited to a few meters ($<10\,\mathrm{m}$) and can be easily disturbed by turbulence, floating sediment, or lighting conditions, which in conjunction with the requisite lighting devices can be power intensive. However, the last decade has witnessed multiple instances of real-world successful implementations, such as Fleischer (2000); Garcia et al. (2001); Eustice et al. (2005); Singh et al. (2007); Johnson-Roberson et al. (2010)).

Generally speaking, in underwater visual SLAM, a number of overlapping pictures of the seafloor are collected and features are identified. From the displacement of features in the images, the trajectory of the AUV can be

estimated, however is prone to drift over time (dead-reckoning). When the vehicle recognize features in the image that are the same with ones from the trajectory history, that means that an area is revisited. The difference between the estimated position of the features and the measured one, is back-propagated and corrects the trajectory traveled, thus closing the loop. At the same time, the registered images are combined to produce a photo-mosaic of the traveled area (Garcia et al., 2006; Mahon et al., 2008; Pizarro et al., 2010; Elibol et al., 2010). A number of oceanographic applications share the requirement for high-resolution imaging of sites extending over hundreds of meters. These include fisheries habitats (Reynolds et al., 2001), coral reefs (Singh et al., 2004), hydrothermal vent sites (German et al., 2004), cold seep sites (Newman et al., 2004), and shipwrecks or archaeological significance sites (Eustice et al., 2005; Foley et al., 2009; Mahon et al., 2011; Gracias et al., 2013).

With the increasing endurance of AUVs, very large areas can be explored, which results in very large photo-mosaic datasets. The management, aligning, and registrations of these datasets are computationally expensive and there significant effort has been expended in this field (Ferrer et al., 2007; Williams et al., 2007; Salvi et al., 2008). There is also a growing interest for the fusion of optical and acoustic data. A notable work from Williams and Mahon (2004) reports an optical camera system that tracks point feature targets initialized by a pencil-beam sonar within the camera's field of view. More recently, the authors Kunz and Singh (2013) presented advances in three-dimensional (3D) mapping of underwater terrain, fusing visual data from a single camera with range data from a multibeam sonar.

Sonar SLAM

Rapid attenuation of the high frequencies in water, makes the use of ultra high resolution devices impractical for longer distances, including vision or laser scanners. Sonar based underwater SLAM utilizes transceivers that work with low frequency sound waves which can penetrate deeper in the water (10-150 m). Sonar sensors are not subject to water visibility issues but they

provide limited information and, medium to low resolution and refresh rates, and can be very expensive. Because of the low resolution, and the acoustic reflexions and shadows it is difficult to robustly extract features and the loop-closure part of the SLAM becomes much harder than with vision. Although there are fewer works in underwater sonar based SLAM than in vision, the research efforts are promising. Underwater sonar commonly employed in AUVs exist in a variety of types, and are therefore they are approached with different methodologies. A review of selected works is provided below. In addition, Table 2.1 at the end of this section provides a summary of different aspects of these works.

Feature based methods Several methods have been reported using features extracted from different types of imaging sonar. Carpenter (1998) presented a simplified EKF implementation in which independent filters were initialized with every new landmark extracted from acoustic images acquired with a high resolution array forward looking sonar. A key aspect of this work is the procedure to obtain the landmarks. In the results section, the estimated trajectories were represented together with the differential global positioning system (DGPS) measurements as ground truth. However, no uncertainty bounds were represented. Using this same data set, a second research was made in collaboration with MIT researchers (Leonard et al., 2001). They presented an EKF based framework which produced a full stochastic map such as correlations were taken into account. Among these introductory works, it is also worth mentioning Leonard and Feder (2001), who presented the decoupled stochastic mapping (DSM), a computationally efficient approximation to large-scale SLAM. At the University of Sydney, Newman (1999) worked on a SLAM framework called the geometrical projection filter (GPF), an approach which estimates the relationships between individual landmarks rather than the location of landmarks in global coordinates. The GPF was tested in a swimming pool. Later, new experiments took place in a real natural terrain along the coast of Sydney, Australia (Williams et al., 2001). This time, a classical EKF implementation of the stochastic map became the core of the SLAM system with artificial landmarks.

At Heriot-Watt University, Tena (2001) followed the path opened by Carpenter (1998) and part of their efforts focused around the study and development of techniques to characterize landmarks from acoustic images. In addition to the point coordinates, a vector of landmark characteristics (including size, perimeter, compactness, maximum dimension, centroid, and invariant moments) was introduced to improve the data association process. In Newman et al. (2003), the synthetic aperture sonar (SAS) developed at MIT, acts as the primary sensor for landmark detection. The paper presents an implementation of the method previously described in Leonard and Newman (2003): the constant time SLAM (CTS). In Maki et al. (2006), a particle filter SLAM framework took advantage of bubble plumes present in the area as well as two sonar reflectors specifically deployed to serve as landmarks. Ship hull inspections can be an important application for AUVs. In this context Walter et al. (2008) describe a SLAM implementation using forward looking sonar (FLS) data from a highly maneuverable, hovering AUV. The exactly sparse extended information filter (ESEIF) algorithm is applied to perform SLAM based upon features manually selected within FLS images.

Side scan sonars provide long range back-scattering information from a profile perpendicular to vehicle motion. Imagery from side scan sonar has been used to address with the underwater SLAM problem in Tena et al. (2004). The method relies on a classical EKF implementation of the stochastic map whose estimated trajectory is then smoothed with a Rauch-Tung-Striebel (RTS) filter and is tested in simulation and with real data. In that work, the stochastic map stores the location of landmarks extracted from the side scan sonar images. This landmark extraction is performed manually along with the data association process. In Tena et al. (2003), the work is extended by addressing the problem of automatic extraction and association of landmark observations. To improve the map consistency from large side scan sonar datasets and reduce computational cost, Aulinas et al. (2010) propose a SLAM approach that uses independent local maps together with a global level stochastic map. The global level contains the relative transformations between local maps.

In the aforementioned works, the extracted features are represented as

points. Using a MSIS, Ribas et al. (2008) represent the walls of a marina environment as line features. A robust voting algorithm has been developed to extract line features, together with their uncertainty, from the continuous sonar data flow. The resulting information is incorporated into a feature-based EKF-SLAM algorithm incorporating sub-mapping techniques.

Although they did not use sonar, the two following works utilize underwater acoustic landmarks in a SLAM framework. In Newman and Leonard (2003), a SLAM framework was proposed to simplify the operation of LBLs by making on-the-fly calibration of submerged transponders using range-only measurements. A different approach to solve the range-only problem is implemented by Olson et al. (2004), in which the authors present an algorithm that imposes geometric constraints on the acoustic measurements to reject outliers.

Featureless based methods Nevertheless, in a natural underwater environment it may be difficult to recognize features via sonar. Their appearance often changes dramatically depending on the point of view, making it extremely difficult to extract robust features. For this reason, a number of researchers have focused their efforts on using featureless methods such as occupancy grids and scan matching.

Fairfield et al. (2007) proposed a method consisting of a Rao-Blackwellized particle filter for which the map is stored within a 3D evidence grid. To reduce the memory requirements, it uses the deferred reference counting octree data structure. It was tested in a tank with an AUV equipped with an array of 54 narrow beam sonar transducers prepared to explore cave systems, and with a 32 beam sonar probe lowered in a flooded natural sinkhole. In ancient water storage systems (cisterns), Clark et al. (2008) tested four different mapping and localization techniques with an MSIS-equipped ROV: Sonar image mosaics using stationary sonar scans, SLAM while the vehicle was in motion, SLAM using stationary sonar scans, and localization using previously created maps. However, the resulting sonar mosaic was produced by manually registering the scan images. In the same expedition, McVicker et al. (2012), used a combination of particle filter and least squares (LS) to estimate the

ROV pose and feed an octree-based occupancy grid map. With the dataset from Ribas et al. (2008), the authors in Burguera et al. (2010) used iterative closest points (ICP) scan matching and an iterative EKF to estimate the trajectory of the AUV.

A multibeam profiler sensor is used in Roman and Singh (2005), who proposed a method to improve bathymetric mapping using SLAM. The solution consisted of generating a set of sub-maps from small bathymetric patches created over short periods of time and cross-registering the overlapping ones with ICP. With the same kind of sensor, the authors in Barkby et al. (2011) used a Rao-Blackwellized particle filter to account for the uncertainty in the vehicle's navigation solution and represent the environment as an elevation map distributed across the ancestry of a given particle. Palomer et al. (2013) proposed a probabilistic implementation of the ICP that addresses the uncertainty of the robot pose as well as the measured points from a multibeam profiler sensor, in a two-stage process including point-to-point and point-to-plane metrics. The authors use octrees for surface adaptation and heuristics based on the uncertainties of the surface points in order to improve the basic algorithm, decreasing the ICP complexity to O(n).

In a different line of research, Pfingsthorn et al. (2010) proposed the use of Fourier-based techniques to register images in a SLAM framework and extend this work later for sonar-based underwater 3D SLAM (Pfingsthorn et al., 2012). Hurtós et al. (2013) proposed also the use of Fourier-based techniques to register two-dimensional (2D) images from FLS multibeam. The estimated registrations are used within a pose-graph optimization framework to obtain a globally consistent trajectory and to render acoustic mosaics of underwater environments.

2.2.3 Thesis approach

There are two main SLAM frameworks: online and full SLAM. The online SLAM estimates most recent pose and map (faster but less accurate) whilst the full SLAM estimates entire path and map (slower but more accurate). Depending the optimization method, a SLAM algorithm may need to batch

process all the information after the experiment has ended (off-line) in order to provide solution or it can process incrementally as new information arrives.

The work presented in this thesis answers the question of how to archive real-time, bounded error, precision navigation in confined underwater environments. We propose a pose-based algorithm to solve the full SLAM problem for AUVs navigating in unknown and possibly unstructured environments with no prior structural information or initial pose consideration. The method first estimates the local path traveled by the robot while forming the acoustic image (scan) with range data coming from an MSIS, providing position estimates for correcting the distortions that the vehicle motion introduces in the scans. Each new pose of a scan is maintained in an ASEKF and is compared with previous scans that are in the nearby area using scan matching. If there enough data is overlapping, a new scan match will put a constraint between the poses updating the ASEKF. These constraints help to identify and close the loops which correct the entire trajectory, thus bounding the drift. The choice of the ASEKF is based on its well studied properties over decades of research, which has been applied successfully in large-scale environments and in real-time applications.

Table 2.1: Summary of selected works on underwater sonar-based SLAM

	Authors	Principal sensor	Landmarks and environment	Filtering	Map	Feature extraction	Data association	Ground truth
feature-based methods	Carpenter (1998)	IS (electr.)	Natural (sea)	EKF (decoupled)	Point features	Automatic	Automatic	Yes
	Newman (1999)	IS (mech.)	Artificial (pool)	EKF (GPF)	Point features	Automatic	Automatic	Yes
	Tena (2001)	IS (electr.)	Natural (sea)	EKF	Point features	Automatic	Automatic	No
	Leonard et al. (2001)	IS (electr.)	Natural (sea)	EKF	Point features	Automatic	Automatic	Yes
	Williams et al. (2001)	IS (mech.)	Mixed (reef)	EKF	Point features	Automatic	Automatic	No
	Leonard and Feder (2001)	IS (mech.)	Artificial (tank)	EKF (DSM)	Point features	Automatic	Automatic	Yes
	Tena et al. (2003)	Sidescan sonar	Natural (sea)	EKF	Point features	Automatic	Automatic	No
	Newman and Leonard (2003)	LBL (range only)	Artificial (sea)	LS optimization	Point features	Automatic	Solved	Yes
	Newman et al. (2003)	IS (SAS)	Natural (sea)	Undefined (CTS)	Point features	Automatic	Manual	Yes
	Olson et al. (2004)	LBL (range only)	Artificial (sea)	EKF	Point features	Automatic	Solved	Yes
	Tena et al. (2004)	Sidescan sonar	Natural (sea)	EKF/RTS	Point features	Manual	Manual	No
	Maki et al. (2006)	Profiler (mech.)	Mixed (sea)	PF	Point features	Automatic	Automatic	No
	Ribas et al. (2008)	IS (mech.)	Artificial (marina)	EKF	Lines features	Automatic	Automatic	Yes
	Walter et al. (2008)	Acoustic camera	Artificial (ship hull)	ESEIF	Sonar mosaic	Automatic	Automatic	Yes
	Johannsson et al. (2010)	Acoustic camera	Mixed (sea/ship hull)	Pose graph	Sonar mosaic	Automatic	Automatic	Yes
featureless methods	Roman and Singh (2005)	Multibeam profiler	Natural (sea)	EKF	Bathymetric	–	Automatic	Yes
	Fairfield et al. (2007)	32 pencil beams	Natural (cenote)	PF	Occupancy grid	–	Automatic	No
	Clark et al. (2008)	IS (mech.)	Natural (cenote)	PF	Occupancy grid	–	Automatic	No
	Burguera et al. (2010)	IS (mech.)	Artificial (marina)	IEKF	Scan points	–	Automatic	Yes
	Barkby et al. (2011)	Multibeam profiler	Natural (sea)	PF/EIF	Bathymetric	–	Automatic	No
	McVicker et al. (2012)	IS (mech.)	Natural (cenote)	PF/LS	Occupancy grid	–	Automatic	No
	Pfingsthorn et al. (2012)	IS (3D)	Mixed (sea/sim)	Pose graph	Cloud of points	–	Automatic	Yes
	Hurtós et al. (2013)	Acoustic camera	Natural (sea)	Pose graph	Sonar mosaic	–	Automatic	Yes

Chapter 3

Working with acoustic images from MSIS

Most mobile robots use laser technology sensors, which have two major advantages in addition their high accuracy range: they gather data for each sector they scan almost instantaneously and the beam angle measurement has almost no uncertainty. However, when using acoustic or ultrasonic range finders these advantages are no longer valid because of the sensors' lower angular resolution and the sparsity of the readings. Underwater vehicles use acoustic sonar instead of laser sensors for long range perception because light and high frequencies propagation in water is very poor (Medwin and Clay, 1998).

Commercially available underwater sonar sensors cover a scan sector by emitting acoustic beams either with a mechanical head that rotates at fixed angular intervals or simultaneously by a multibeam sonar head. The new multibeam forward looking sonar sensors recently introduced in the market have improved resolution and refresh rates. However, relatively few of them have been deployed in the field and their scan sectors are limited to around 120 degrees.

On the other hand, MSIS has been a very popular choice for many years, offering full 360 deg scan sectors (Fig. 3.1), at a relatively small size and low cost. Its main drawbacks, which we will address in this section, are its noisy

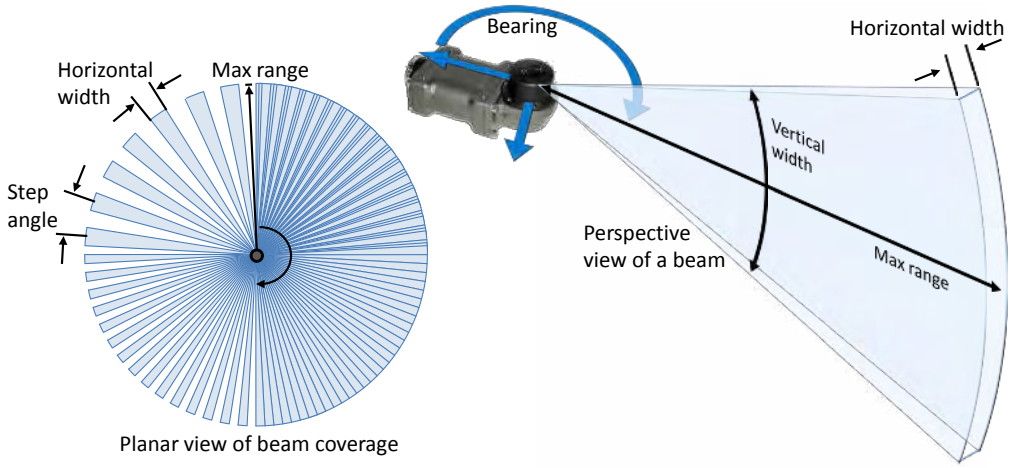


Figure 3.1: MSIS acoustic coverage.

measurements and its relatively slow rotating mechanical heads. At each step, a beam is emitted and received, measuring the intensities of the acoustic reflections from the obstacles found across its path (Fig. 3.2). Acquiring a scan for a complete 360 deg sector takes several seconds, during which the motion of the vehicle causes distortions in the final acoustic image (Fig. 3.3). For this reason, it is necessary to take into account the robot's trajectory while the beams of a scan sector are being acquired.

To address these problems, we introduce the *ScanGrabbing* (Algorithm 1) whose role is to collect all the beams that form a full 360 deg sonar scan sector (herein referred to as a *scan*), filter them to remove noise, and recover the correct final scan image from the distortions suffered due to vehicle motion.

The following three sections present the three major parts of the *ScanGrabbing* algorithm:

1. relative vehicle localization,
2. beam segmentation, and
3. scan forming.

Algorithm 1 ScanGrabbing algorithm

$[S_{new}, \hat{\mathbf{q}}_{new}, \mathbf{P}_{q_{new}}] = ScanGrabbing()$ {
  $ResetDeadReckoningXYZ()$
  $[\hat{\mathbf{x}}_B^I, \mathbf{P}_{IB}] = GetDeadReckoning()$
  for all beams {
    $beam = GetBeam()$
    $beam = Segment(beam)$
    $[\hat{\rho}, \mathbf{P}_\rho] = LocalMaximaFinder(beam)$
    $[\hat{\mathbf{x}}_R^B, \mathbf{P}_{BR}] = GetDeadReckoning()$
    $//\hat{\rho}$ and $\mathbf{P}_\rho$ from the local frame I
    $[\hat{\mathbf{z}}_i^{I_c}, \mathbf{P}_{z_i}] = ScanForming(\hat{\mathbf{x}}_R^B, \mathbf{P}_{BR}, \hat{\rho}, \mathbf{P}_\rho, I_c)$
    $S_{new} = S_{new} \cup \{[\hat{\mathbf{z}}_i^{I_c}, \mathbf{P}_{z_i}]\}$
  }
  $\hat{\mathbf{q}}_{new} = \hat{\mathbf{x}}_R^B$
  $\mathbf{P}_{q_{new}} = \mathbf{P}_{BR}$
}

3.1 Relative vehicle localization

Since MSIS needs a considerable amount of time to obtain a complete scan (around 5∼20 sec, depending on the resolution and range settings), motion induces a distortion in the acoustic image when the vehicle does not remain static, which is very common in water (Fig. 3.3). To address this problem, it is necessary to know the vehicle's pose at both the beam transmission and reception time. However, the time difference between beam transmission and reception can be neglected for two reasons. First, the practical operational range of the sonar does not usually exceed 100 m such that with the mean speed of sound in sea water of 1,500 m/sec, the total two-way travel time of the acoustic signal is approximately 0.07 sec. Second, most of the AUVs are moving at relatively low speeds, especially when they are expected to avoid obstacles, so the motion of the vehicle between beam transmission and reception time can be assumed to be practically zero, and only the reception position needs to be considered. Therefore, we define an initial coordinate system I_c to reference all the range measurements belonging to the same scan. In order to reduce the influence of the motion uncertainties to the scan, as Burguera et al. (2008) suggested, we set this reference frame at the robot's

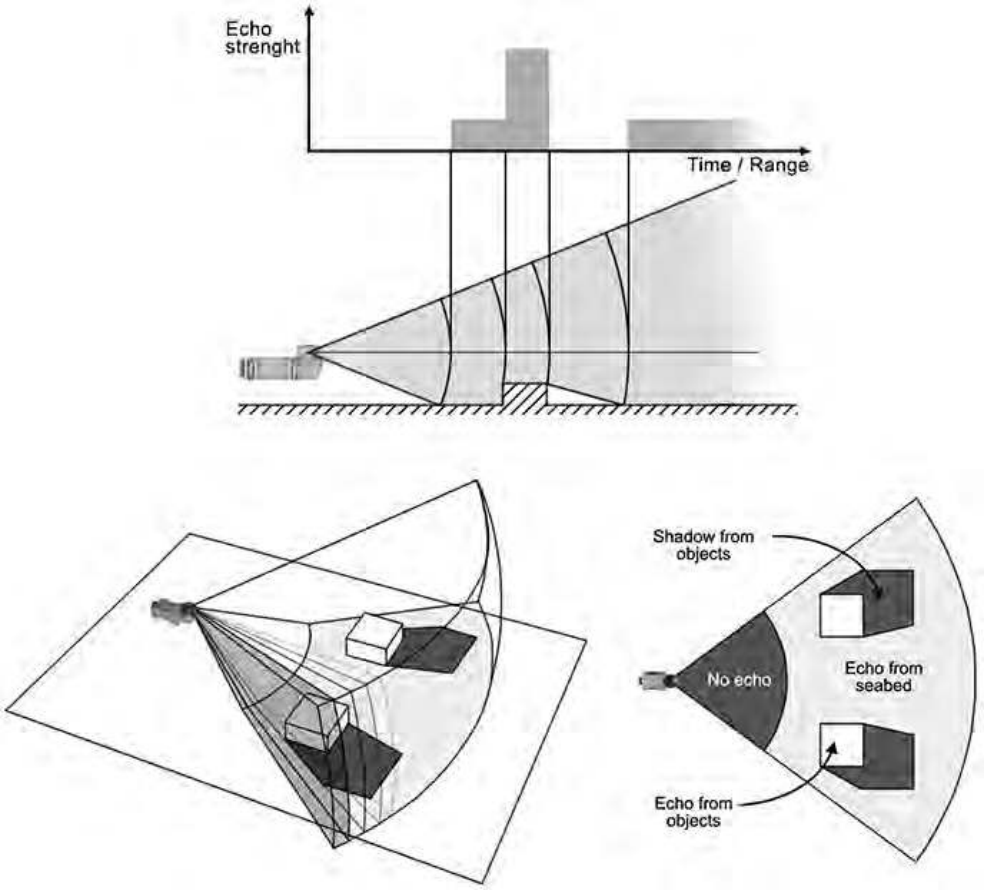
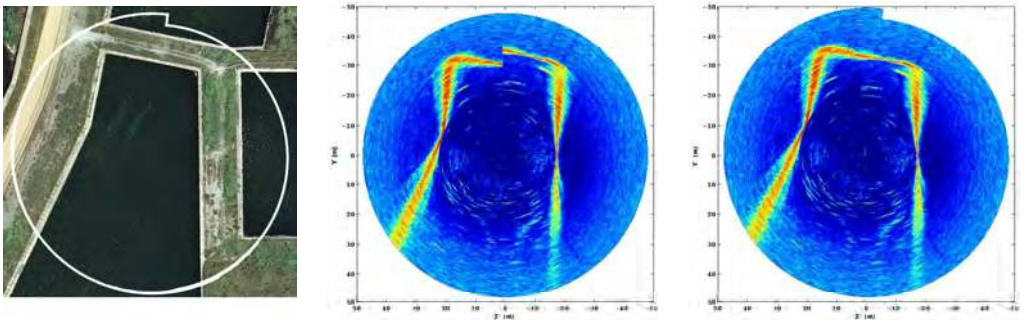


Figure 3.2: In top: Generation of an acoustic beam. Bottom: Scanning a sector to produce a sonar image (Ribas et al., 2008).

pose where the center beam of the current scan was received.

The localization system used herein to estimate vehicle motion follows the navigation system described in Ribas et al. (2008). In this system, an attitude and heading reference system (AHRS) provides attitude measurements and a DVL unit including a pressure sensor are used to estimate the robot's velocity and depth during the scan. All measurements occur asynchronously with the MSIS beams arriving at a 30 Hz rate while the DVL and AHRS readings arrive with a frequency of 1.5 Hz and 10 Hz respectively. The AUVs used in this work are very stable in roll and pitch and during the experiments they performed survey patterns at almost constant speed. Therefore, a simple four degrees of freedom (DOF) constant velocity kinematic model is used to



(a) (b) (c)

Figure 3.3: Effect of motion induced-distortion on acoustic images: (a) Orthophoto map of the real environment where sonar data was gathered. (b) The sonar data represented in Cartesian coordinates. (c) Undistorted image obtained after taking into account vehicle motion.

predict vehicle motion. An EKF is used to estimate the robot's pose whenever a sonar beam is received and to update the model prediction each time a new DVL or AHRS measurement arrives. An analytical description of the system model is presented below.

The information of the system at step k is stored in the state vector $\mathbf{x}_k$ with estimated mean $\hat{\mathbf{x}}_k$ and covariance $\mathbf{P}_k$:

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \eta^B \\ \nu^R \end{bmatrix}, \quad \mathbf{P}_k = E\left[\left(\mathbf{x}_k - \hat{\mathbf{x}}_k\right)\left(\mathbf{x}_k - \hat{\mathbf{x}}_k\right)^T\right] \tag{3.1}$$

with:

$$\eta^B = [x, y, z, \psi]^T, \quad \nu^R = [u, v, w, r]^T \tag{3.2}$$

where, as defined in Fossen (1994), η^B is the position (x, y, z) and attitude (ψ) vector referenced to a base frame B and ν^R is the linear (u, v, w) and angular (r) velocity vector referenced to the robot's coordinate frame R. The base frame B is chosen coincident with I but aligned with a North heading to facilitate the integration of compass measurements. Finally, the model prediction and update are carried out as described in the next section.

3.1.1 Prediction

The vehicle's movement prediction at step k with a time step Δt is performed using the following 4 DOF constant velocity kinematic model:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, n_k) \longrightarrow \begin{bmatrix} \eta_k^B \\ \nu_k^R \end{bmatrix} = \begin{bmatrix} \eta_{k-1}^B + R(\eta_{k-1}^B)(\nu_{k-1}^R \Delta t + n_k \frac{\Delta t^2}{2}) \\ \nu_{k-1}^R + n_k \Delta t \end{bmatrix} \tag{3.3}$$

with:

$$R(\eta) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 & 0 \\ sin(\psi) & cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

In this model the velocity is considered to be constant with a velocity perturbation modeled as the integral of a zero-mean white Gaussian noise n_k. The covariance matrix $\mathbf{Q}_k$ of this acceleration noise is assumed diagonal and of the same order of magnitude as the maximum acceleration increment that the robot may experience over a sample period.

The estimate of the state is obtained with the standard EKF prediction equations, hence, n_k is an acceleration noise integrated and added to the velocity (3.3), which is then non-linearly propagated to the position.

The estimate of the state is obtained as:

$$\hat{\mathbf{x}}_k = f(\hat{\mathbf{x}}_{k-1}) \tag{3.5}$$

and its covariance matrix as:

$$\mathbf{P}_k = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T \tag{3.6}$$

where $\mathbf{F}_k$ and $\mathbf{G}_k$ are the Jacobian matrices of partial derivatives of the non-linear model function f with respect to the state $\mathbf{x}_k^B$ and the noise n_k, respectively.

3.1.2 Update using DVL or AHRS measurements

Whenever new velocities or heading measurements are available from the DVL or AHRS sensor, the model prediction is updated using the standard Kalman filter equations:

$$\mathbf{z}_{DVL,k} = [u_{b,w}, v_{b,w}, w_{b,w}, z_{depth}]^T, \qquad \mathbf{z}_{AHRS,k} = [\psi] \tag{3.7}$$

where subindices b or w stand for bottom tracking velocity or water layer tracking velocity, respectively. The measurement model becomes:

$$\mathbf{z}_{DVL,k} = \mathbf{H}_{DVL}\mathbf{x}_k + w_k, \qquad \mathbf{z}_{AHRS,k} = \mathbf{H}_{AHRS}\mathbf{x}_k + w_k \tag{3.8}$$

$$\mathbf{H}_{DVL} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{3.9}$$

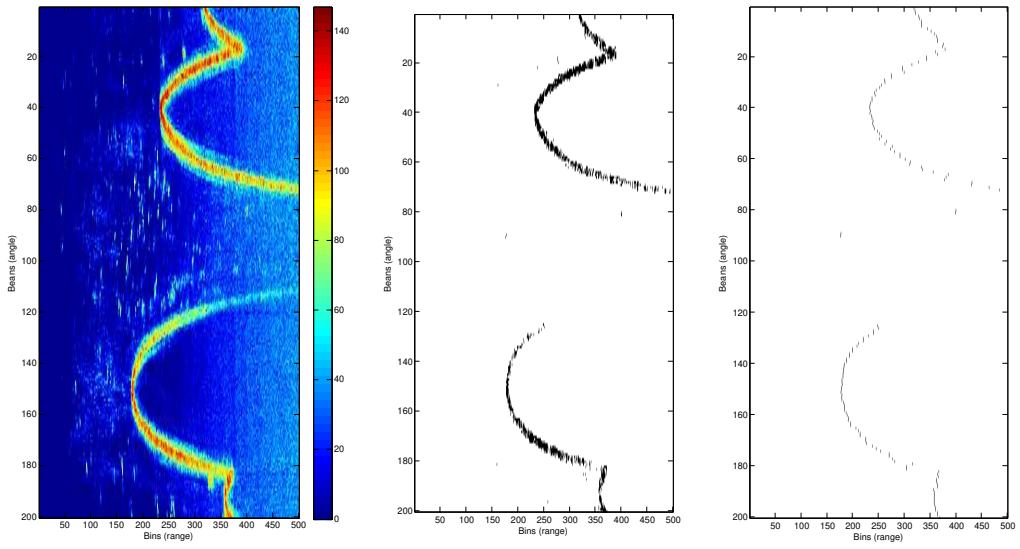
$$\mathbf{H}_{AHRS} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where w_k (measurement noise) is a zero-mean white Gaussian noise.

3.2 Beam segmentation and range detection

We can represent a full scan as a polar acoustic image composed of beams (Fig. 3.4a). Each beam has a particular bearing angle value with respect to the MSIS sensor and a set of intensity measurements acquired at known intervals along the beam path. The angle corresponds to the orientation of the sensor head when the beam is emitted. The acoustic linear image corresponding to one beam is returned as an array of acoustic intensities (bins) detected at a certain distance.

Due to the noisy nature of the acoustic data, the acquisition of a range



(a) (b) (c)

Figure 3.4: MSIS beam segmentation: (a) Raw data from Fig. 3.3 represented in polar coordinates. (b) Data after applying a threshold. (c) Selection of the local maxima bins for each beam.

measurement for the beam is segmented in two steps. First, only those bins with an intensity value over a threshold are selected and stored. This procedure separates the acoustic imprint left by an object in the image from the noisy background data (Fig. 3.4b). The second step is to select among the beams for which the threshold is exceeded the bins that are local maxima and satisfy a minimum distance criterion between them. This means that if two or more of these bins are too close within the beam, they correspond to the detection of the same object and are redundant. Then, the bins with the lowest intensity value are discarded (see the result in Fig. 3.4c). The selected local high-intensity bins are the ones that most likely correspond to objects present in the scene.

3.3 Scan forming

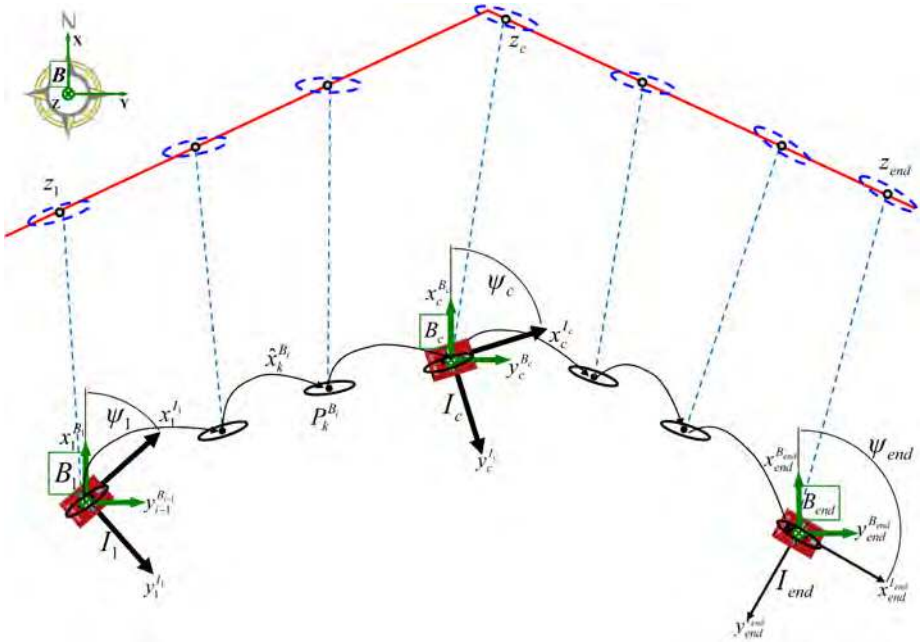
The navigation system presented above is able to estimate the robot's pose, but the uncertainty can grow without limit due to its dead-reckoning nature. However, we are only interested in the robot's relative position and uncertainty within a scan with respect to the center of that scan (I_c frame, Fig. 3.5). For this reason the filter starts with the position and uncertainty of zero whenever a new beam is emitted, which keeps the consecutive beam poses statistically independent.

The filter is initialized with the yaw (ψ) value from the attitude sensor because it represents an absolute angle with respect to the magnetic North. This enables, the modified filter to provide the robot's relative displacements (and their uncertainties) between consecutive beam acquisition poses. With these available, measurements can be combined with vehicle motion to reduce the effect of distortions, while referencing them to the new frame I_c to produce a better uncertainty distribution (Burguera et al. (2008)).

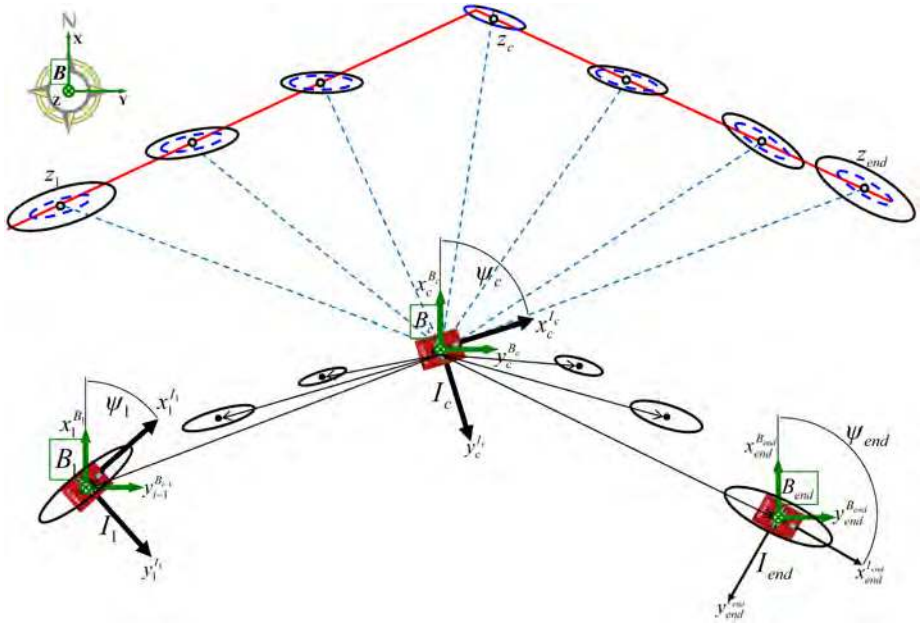
It is very common in mobile robotics for a dead-reckoning system to provide the relative angle and displacement between consecutive poses together with their uncertainties. It is then straightforward to construct a trajectory from several consecutive translations and to reference them to any desired frame by means of compounding or inverse transformations, as described in Smith et al. (1990) and applied in Burguera et al. (2008).

However, absolute compass readings are not in the same reference frame as the corresponding relative displacements. Therefor, we propose a different approach to forming a scan with respect to a common coordinate frame I_c, as described below. Let,

- $\rho_i^{I_i} \equiv N(\hat{\rho}_i^{I_i}, \mathbf{P}_{\rho_i})$ be the measurement points in polar space (as they provided by the sensor) modeled as Gaussian random variables (GRVs), of the scan to be formed in the I_i frame corresponding to the related robot pose at step i,
- $(\hat{\mathbf{z}}_i^{I_i}, \mathbf{P}_{z_i}) = P2C(\hat{\rho}_i^{I_i}, \mathbf{P}_{\rho_i})$ be these measurement points converted to the Cartesian space,



(a) Each vehicle pose corresponds to the acquisition of a beam and is represented with respect to a new North-aligned reference frame. The arcs between poses represent the relative displacements.



(b) The beams in a full scan are then referenced to the I_c frame, which corresponds to the position at the center of the trajectory. The uncertainty of the dead-reckoning process is distributed accordingly and propagated to the scan points.

Figure 3.5: The scan forming process, simplified for clarity.

- B_i be the North-aligned frame sharing the same origin as I_i,
- $\mathbf{x}_i^{B_i} \equiv N(\hat{\mathbf{x}}_i^{B_i}, \mathbf{P}_{x_i})$ be the robot state and uncertainty from the dead-reckoning EKF, with $\hat{\mathbf{x}}_i^{B_i} = [x_i, y_i, \psi_i]^T$,
- $\mathbf{d}_i^{B_i} \equiv N(\hat{\mathbf{d}}_i^{B_i}, \mathbf{P}_{d_i})$ be the robot displacement between time step $i-1$ and i, where $\hat{\mathbf{d}}_i^{B_i} = [x_i, y_i, 0]^T$ are elements from the robot state $\hat{\mathbf{x}}_i^{B_i}$ and $\mathbf{P}_{d_i}$ the corresponding submatrix from its covariance matrix, and
- $\mathbf{r}_i^{B_i} \equiv N(\hat{\mathbf{r}}_i^{B_i}, \mathbf{P}_{r_i})$ be the rotation transformation between the robot coordinate frame I_i with the corresponding North-aligned frame B_i, where $\hat{\mathbf{r}}_{I_i}^{B_i} = [0, 0, \psi_i]^T$ are elements from the robot state $\hat{\mathbf{x}}_i^{B_i}$ and $\mathbf{P}_{r_i}$ the corresponding submatrix from its covariance matrix,

then, we can form the scan with respect to the center frame I_c (Fig. 3.5), as follows:

$$\mathbf{z}_i^{I_c} = \begin{cases} \mathbf{z}_i & i = c \\ \ominus \mathbf{r}_{I_c}^{B_c} \oplus \mathbf{D}_{I_c}^{B_c} \oplus \mathbf{r}_{I_i}^{B_i} \oplus \mathbf{z}_i^{I_i} & i \neq c \end{cases} \tag{3.10}$$

where

$$\mathbf{D}_{I_c}^{B_c} = \left\{ \begin{array}{ll} \mathbf{D}^+ = \left(+\mathbf{d}_c^{B_c} + \ldots + \mathbf{d}_{i-1}^{B_{i-1}}\right) & i > c \\ \mathbf{D}^- = \left(-\mathbf{d}_c^{B_c} - \ldots - \mathbf{d}_{i-1}^{B_{i-1}}\right) & i < c \end{array} \right\} = \sum_{j=c}^{i-1} sign\,(i-c)\,\mathbf{d}_j^{B_j} \tag{3.11}$$

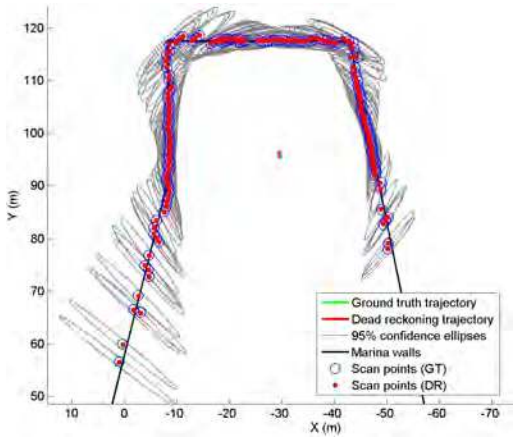
is the relative robot pose referenced to B_c where the point $\mathbf{z}_i$ was observed. Because all the consecutive robot displacements $\mathbf{d}_j^{B_j}$ are represented in a North-aligned frame (B_j), a simple linear vector addition operation is sufficient to compute $\mathbf{D}_{I_c}^{B_c}$.

The uncertainty $\mathbf{P}_{z_i}$ of the scan points $\hat{\mathbf{z}}_i^{I_i}$ can be easily computed by using the Jacobian matrices of the compounding and inverse transformations (Smith et al., 1990) and applying them to Equation 3.10.

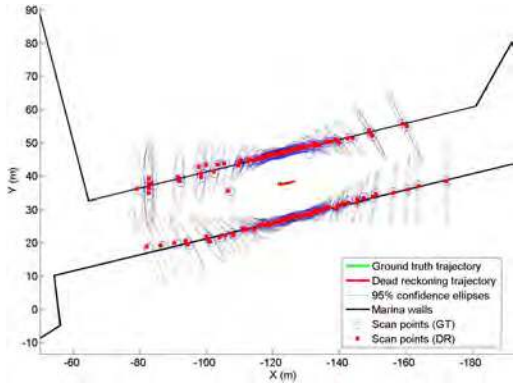
Figures 3.6a and 3.6b illustrate the results of the scan-forming algorithm by showing as an example two scans from a real environment. The readings shown correspond to the acoustic images in Figure 3.3 and to the corridor part of the marina dataset (see Section 7.4). The reference point I_c of each scan has been placed at the corresponding ground truth position in order

to highlight the coherence of the scans compared with the ground truth trajectory. The scan trajectory of Fig. 3.6b is shown in more detail in 3.6c with the 95 % confidence ellipses.

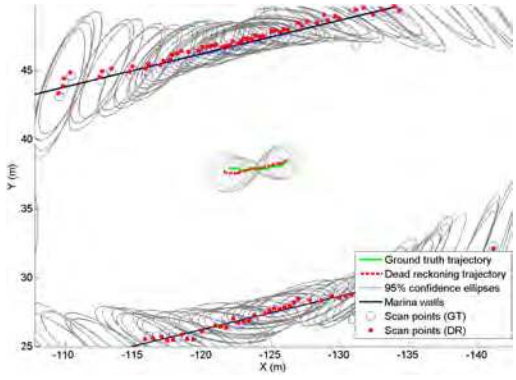
It can be easily seen that the proposed scan forming algorithm builds the scan in agreement with the ground truth and that the absolute drift of the dead-reckoning system of the vehicle during the scan is small due to the short time required for a complete scan. The histogram of the absolute error of all the scans is shown in the results section (Fig. 7.8) and is comparable to the DGPS error that we use as a ground truth. For this reason, the deformations still remaining after the composition of the measurements with the dead-reckoning trajectory can be assumed to be negligible for short trajectories. Therefore, once created, the scan is considered to be rigid and it is not modified during the SLAM process.



(a)



(b)



(c)

Figure 3.6: Scan forming examples over their corresponding ground truth reference positions: (a) The scan from Fig. 3.3. (b) Scan of a corridor from the same environment. (c) Magnification showing the trajectory of the dead-reckoning and the ground truth.

Chapter 4

Scan Matching

The term *scan matching* refers to a family of algorithms that given two partially overlapping scans, can identify the rigid-body transformation that aligns them best. The goal of scan matching in robotics is to compute the relative displacement of a vehicle between two scans, or a map and a scan, by maximizing the overlap between the range measurements obtained from a laser or a sonar sensor. Given a reference scan S_{ref}, a new scan S_{new} and an initial guess of the displacement estimation $\hat{\mathbf{q}}$ between them, the objective of scan matching methods is to obtain a better estimate of the real displacement $\mathbf{q} = (x, y, \theta)$ (Figure 4.1).

The scan matching problem has been addressed in different ways, but the 2D problem has a consistent body of literature. The existing algorithms can be divided in four groups of methods according to their nature:

- ICP methods,
- Monte Carlo methods,
- Correlation methods, and
- Newton's methods

The following sections give a synopsis of the literature for the main contributions within the groups identified above. These are summarized in Table 4.2 at the end of the section.

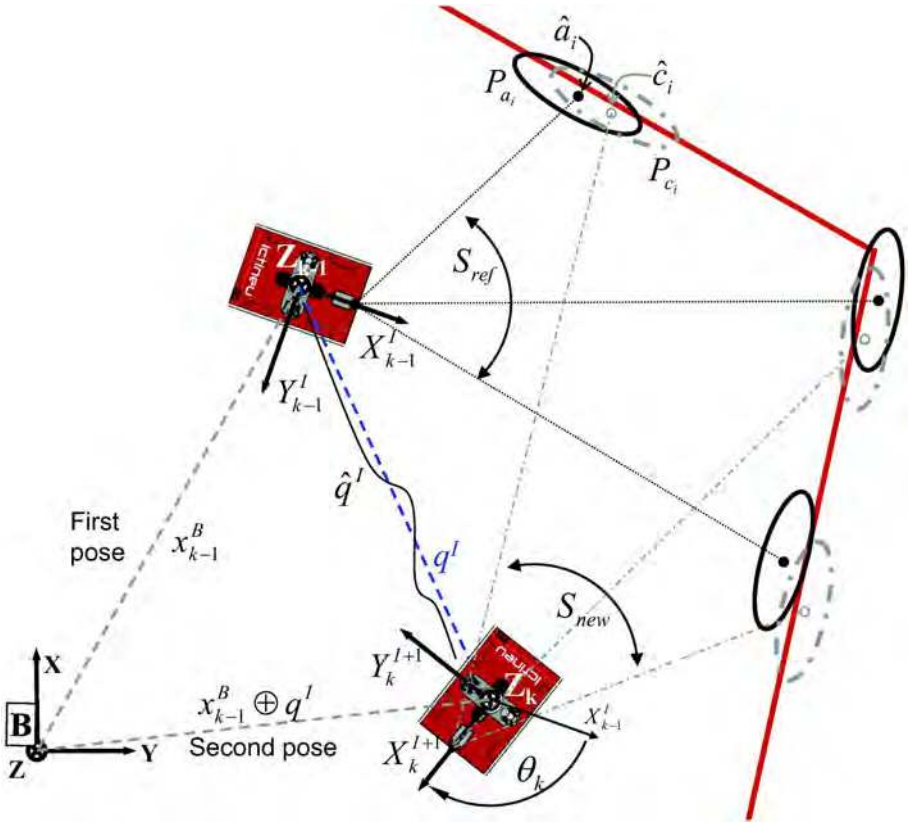


Figure 4.1: Scan matching problem description.

4.1 ICP methods

The ICP algorithm from Besl and McKay (1992), originally developed for the registration of 3D images, is expressed as a cloud of points for computer vision purposes, and addresses the scan matching problem using an iterative process that follows two steps. First, the correspondences between scans are computed using the Euclidean distance of each point of the new scan with the nearest point of the reference scan based on an initial displacement. Next, the mean squared error cost function is applied in order to minimize this distance through rotation and translation, that will be the new displacement estimate. The process is repeated again but the estimation obtained in the previous iteration is used as the prior displacement. Iterations continue until the solution converges to the nearest local minimum.

Algorithm 2 General ICP algorithm

1: **procedure** $ICP(S_{ref}, S_{new}, q)$
2: **for** $i = 1$ **to** $n\_iterations$ **do**
3: $C = AssociationSet(S_{ref}, S_{new} \oplus q)$
4: $[q] = Register(C, S_{ref})$
5: **if** $Convergence$ **then return** q
6: **end for**
7: **end procedure**

The algorithm uses the Euclidean distance to select the correspondences and the least squares minimization. This is a limitation because this distance does not take into account the sensor's rotation. To overcome this problem, Lu and Milios (1994) proposed the iterative dual correspondence (IDC) algorithm which computes two sets of correspondences; one using the Euclidean distance to estimate the translation between scans, and another using the matching-range-point rule to reveal the rotation component. However, two minimization processes are required for each iteration, increasing the computational cost of the algorithm. Moreover, the final solution is a composition of the two minimizations. The authors apply it with 2D laser scans and it is considered one of the first implementations for mobile robots. The general structure of an ICP-based scan matching algorithm is presented in Algorithm 2.

Many other variations have been developed to solve more specific problems or to improve the original approach. Trimmed ICP (Chetverikov et al., 2002) uses the least trimmed squares method which constantly cuts down the number of associations by sorting them by their residuals. Picky ICP (Zinßer et al., 2003) tries to mix different minor improvements in either speed or robustness. It mixes RICP (Trucco et al., 1999) which uses the least median square as error function, and MICP by (Masuda and Yokoya, 1994) which uses a random sampling as association control points (similar to the widely known RANSAC). An alternative solution is to take into account the sensor rotation in addition to the translations. In this sense, Mínguez et al. (2005) proposed the metric-based iterative closest point (MbICP) where a new concept of distance uses translation and rotation simultaneously. Introducing this new

distance into the ICP framework, transitional and rotational movements are taken into account simultaneously while looking for correspondences as well as during the minimization process. As a consequence, robustness, accuracy, and convergence are increased significantly. Diosi and Kleeman (2005) worked on a polar version of ICP (working purely with range data) using an iterative minimization approach which they qualify as more robust. This helps keep the number of iterations low and improve the quality of displacement estimates. Censi (2008) used the information provided by the normal of the surface and proposed the point-to-line metric, which converged quadratically in a finite number of steps.

However, geometric representations of the scan matching problem do not allow for modeling of the uncertainty present in the sensor measurements. On the other hand, statistical representations allow for the introduction of scan noise (of both the sensor data and the displacement) in a natural way. Montesano et al. (2005) proposed the probabilistic iterative correspondence (pIC) algorithm which is a statistical extension of the ICP algorithm. To look for statistically compatible associations, the algorithm takes into consideration all the involved uncertainties. Whereas geometric ICP representations use the Euclidean distance to find the correspondence for the points between the scans, the pIC algorithm creates a virtual association from the statistically compatible points using the distance proposed by Mahalanobis (1936). For mobile robots equipped with conventional time-of-fly ultrasonic range finders, Burguera et al. (2007) proposed the sonar probabilistic iterative correspondence (spIC) which is a variation of the pIC algorithm. This method manages the sparsity of readings by grouping sonar readings along short trajectories. It uses probabilistic models of ultrasonic and dead-reckoning sensors as well as a method to propagate the error through them in order to estimate a group of scan positions together with their uncertainty. Then, given a scan of positions referenced to the same frame with their corresponding uncertainties, the procedure followed is similar to the one used by the pIC.

The above algorithms have in common that they use sensors based on laser technology that gathers each scan almost instantaneously or on ultrasonic range finders on rigid mobile platforms. However, for the underwater

environment, commercially available scan sensors are based on acoustics and the motion of the robot is also subject to currents. Usually, the forward looking sensors that can scan a 360 deg sector have a mechanical head that rotates at fixed angular intervals, and a complete scan can take a few seconds. During that time, the vehicle moves resulting deformed scans. To compensate for the issue, a correction is necessary, taking into account the position of the robot when each beam is grabbed. For the underwater domain Hernández et al. (2009) proposed the mechanically scanned imaging sonar probabilistic iterative correspondence (MSISpIC) algorithm, which is an extension of the spIC and pIC. The MSISpIC algorithm forms a scan corrected for the AUV's motion distortion, taking into account the involved uncertainties. After the corrected scan has formed, the algorithm selects two consecutive scans and registers them using a modified spIC algorithm.

The performance of the ICP algorithms family is heavily dependent on the sensor used and the environment in which it is applied, which explain the hundreds of variations for the algorithm available in the literature. This section, review the most famous and original algorithms developed after the first algorithm was developed by Besl and McKay (1992). Interesting classifications and performance comparisons for the most common of them can be found in Gutmann and Schlegel (1996), Rusinkiewicz and Levoy (2001) and François et al. (2013).

4.2 Monte Carlo methods

These methods define an uncertainty area around the robot's pose, which represents the search space for a corrected pose. Once the search space is defined, the algorithms search in different ways over this space for a solution. The decision is made using a heuristic value usually calculated using a grid. The reference scan is mapped into a grid with variable resolution, usually depends on the size of the area covered by the scans. Then, after applying a possible correction to the second scan, that scan is also mapped into that same grid. The number of overlapping cells between the first and the second scan is calculated which represents the score value of the correction. Algorithm 3

Algorithm 3 General Montecarlo algorithm

1: **procedure** $Montecarlo(S_{ref}, S_{new}, q)$
2: $Grid = makeOccupancyGrid(S_{ref})$
3: **for** $i = 1$ **to** $n\_iterations$ **do**
4: $Q = GeneratePoses(q, \delta_d)$
5: $q = argmin_{q_j \in Q}\ fitness(q_j, S_{new}, Grid)$
6: if Convergence **then return** q
7: **end for**
8: **end procedure**

presents the general idea behind the method.

Montemerlo et al. (2003) presented the VASCO scan matcher in the CARMEN framework. The idea is to search for a better pose in the borders of the axes of an uncertainty ellipse. The axes of the ellipse have fixed length and at every iteration the axis length is divided by two. Eliazar and Parr (2004) propose an algorithm that works similarly to VASCO but instead of searching over the axis of the ellipse, it generates a set of possible poses inside the uncertainty ellipse and then assigns a score using a grid. Zandara and Nicholson (2009) redefined the ellipse size using the pose uncertainty thus reducing the computational time by skipping unlikely poses.

Other approaches investigated the generation of poses in a manner similar to genetic evolution. At every iteration, multiple poses are generated evolving from the previous set of poses. The poses that do not have a certain score are discarded. The discarded poses are replaced with what are considered genetic variations of the best poses. Genetic algorithms follow a common implementation which is to make genes evolve by switching their chromosomes. At each iteration the genes with higher scores are mixed together. This aims to keep the best component of the genes, which in our case are the components of the pose. The mixing technique depends on the implementation. Duckett (2003) introduce the genetic algorithm to a SLAM problem using laser scans. Martínez et al. (2006) and Ze-Su et al. (2007) worked both on a mixed ICP algorithm aided by a genetic algorithm. A genetic algorithm is used to search for a suitable coarse correction using the dead-reckoning, after which ICP

is used as the last step to produce a fine correction. Lenac et al. (2007) minimized an alignment error via a genetic algorithm. They proposed an alignment metric based on a look-up table and Gray coding for the scan matching parameters.

4.3 Correlation methods

These methods take advantage of the cross correlation function and return a series of hypotheses based on a signal shift. Such a function is a powerful method for retrieving a shift in signals. The scans are reshaped in order to find a translational invariant signal (1D or 2D) using different methods. Their computational cost depends on the number of hypotheses generated, which is determined before the execution. Algorithm 4 presents the general idea for the classic approach methods.

Weiß and von Puttkamer (1995) reshaped the scans into 1D signals generated using the consecutive points' relative orientation. This information is invariant with translation so a rotation hypothesis is found. Then, the scan is rotated according to that hypothesis and two other histograms are created (x and y histograms) that describe the distribution of the points into the x and y space. Comparing the rotated scan histograms with the reference scan, a translational displacement is found. Bosse and Roberts (2007) enhanced the algorithm by relaxing the assumption of structured environment and demonstrated that their method works well in unstructured outdoor environments.

Following another method, Censi et al. (2005) computed the discrete Hough transform from both scans to be matched and calculated the spectra of the result of the Hough transform. The spectra are then cross-correlated to identify a series of hypotheses on the rotational displacement. To estimate the translation, an alignment direction is selected by searching for the local maximum of the Hough spectrum in the new scan. The Hough spectrum of the new scan is rotated by the hypothesis and then cross-correlated with the hypothesis of the reference scan. Again the local maximum of the last correlation is used to search for the translational hypotheses. Each

Algorithm 4 Classical Cross-Correlation algorithm

procedure $Correlation(S_{ref}, S_{new}, q)$
  $Inva_{ref} = makesignal(S_{ref})$
  $Inva_{new} = makesignal(S_{new} \oplus q)$
  $CC = cross\_correlate(Inva_{ref}, Inva_{new})$
  $M = localMaxima(CC, N_\phi)$
  for $\phi \in M$ **do**
    $SR_{new} = (S_{new} \oplus q) \oplus \phi$;
    $hyp = extract\_hypothesis(S_{ref}, SR_{new})$
    $q = [\phi\, argmax_{(t_i \in hyp)} \{score(SR_{new} \oplus t_i, S_{ref})\}]$
  end for
  return q
end procedure

translational hypothesis is then weighted and the best one is chosen as a final correction.

Another group of correlation algorithms are based on the principle of phase correlation, which states that the shift between two signals is transformed in the Fourier domain into a linear phase shift. This principle has been extended (Chen et al., 1994; Reddy and Chatterji, 1996) to estimate the rotation and scaling between signals by combining a polar-log mapping of the spectral magnitude (corresponding to the Fourier-Mellin transform (FMT)) and phase correlation. This approach has been used in the context of underwater photomosaicing (Pizarro et al., 2001). In the same application field, Bülow et al. (2009) presented a variant of the FMT (named iFMT) in which a polar-logarithmic re-sampling of the images is used to convert rotation and scaling into corresponding phase shift and allow for a registration in one step. The same method has been applied in Bülow et al. (2010) to register underwater scans from a 2D imaging sonar. The correlation between the magnitude spectra of the two scan images in the polar domain is used to find a rotational estimate. Translational estimates are then found by correlating the 2D spectra of the rotational corrected scan and of the reference scan. The estimates are weighted using a score function to choose the best one. An heuristic to express the matching uncertainty of the iFMT method is

introduced in Pfingsthorn et al. (2010) with the aim to embed the registration method in a SLAM framework.

Using a similar method, Hurtós et al. (2013) propose the use of Fourier-based techniques to register 2D FLS images. The registration technique handles similarity transforms in the image alignment (translation and rotation) by first recovering an estimate of the rotation directly from the polar version of the sonar images and then estimating the translation using the rotation compensated image. Unlike FMT approaches, the polar magnitude of the spectrum is not used to estimate the rotation as it is more sensitive to the noise of the sonar images. The author's report that although this strategy does not uncouple rotation from translation and can lead to inaccuracies, it still produces better results for the particular case of registering FLS frames. The estimated transforms are then used within a pose-graph optimization framework to obtain a globally consistent trajectory and render sonar mosaics of underwater environments.

Spectral registration has also been applied to the registration of 3D sonar scans (Bülow and Birk, 2011). In this case the authors present a full 6 DOF registration method which uses phase matching on re-sampled data. As in other frequency-based methods, the algorithm is based on decoupling 3D rotation from 3D translation but in this case this is done by a re-sampling process of the spectral magnitude of a 3D fast Fourier transform, computed on discretized 3D data. The re-sampling scheme allows to process, by phase-correlation, a complete stack of layers in one step instead of processing spherical layers of the 3D spectrum. The method, named as spectral registration with multilayer re-sampling (SRMR), has proved robust against noise and low overlaps and has shown successful results in 3D underwater sonar (Bülow and Birk, 2011) and optical (Bülow et al., 2013) mapping as well as in the registration of laser scans (Bülow and Birk, 2013). Besides, an uncertainty estimation has also been introduced Pfingsthorn et al. (2012) in order to use the 6 DOF spectral registration method inside a sonar-based underwater SLAM framework .

4.4 Newton's methods

First defined by Biber and Straßer (2003), normal distributions transform (NDT) is presented as a new original method for solving the scan matching problem. The two main algorithms, which are so far the only methods based on such a technique for 2D scans, have common features. They use Newton's method to minimize the error function. Newton's method searches for the zeros of a continuous function, and when used in its derivative form, it can find the minimum of a function. Magnusson et al. (2007) ported the NDT to 3D. The general idea is presented in Algorithm 5.

In Biber and Straßer (2003), a likelihood grid is created in which each point in the reference scan is inserted. Then, for every cell with more than five points a mean and covariance are calculated. For every point in the new scan, a corresponding mean and covariance is assigned using a direct lookup in the grid. The error function is then the sum of the error between each point in the new scan and its associated mean and covariance. Thrun (2001) defined the likelihood fields for this problem. This algorithm has similar principles to the previous one but it does not work with grids. Likelihood fields are defined using the two scans. This time, the minimizing function is defined as the exponential of the distances between a point in the new scan and a set of points in the reference scan. The set of points in the reference scan is identified using an Euclidean distance threshold.

Likelihood fields and NDT variations for sonar were presented by Burguera et al. (2009). The likelihood field with sum of Gaussians (LF/SoG) is an LF-based algorithm which approximates one of the scans by a sum of Gaussians. That way, it builds a continuous and differentiable likelihood field. The authors reported improvements in robustness and accuracy in the performance of sonar scan matching.

Algorithm 5 Newton's method to search the correction

```
1: procedure Newton(S_ref, S_new, q)
2:     f = error_function(application based)
3:     for i = 1 to n_iterations do
4:       for all p_i ∈ S_new do
5:         g = g + f'(S_ref, p_i ⊕ q)
6:         H = H + f''(S_ref, p_i ⊕ q)
7:       end for
8:       q = q − H^{-1} g
9:       if Convergence then return q
10:    end for
11: end procedure
```

4.5 Discussion

The above literature review of the major scan matching methods reveals the existence of a large number of algorithms. The reason for that can be traced to the implementation nature of scan matching. In the same environment, different types of sensors will produce different scans as they are characterized from many factors such as sensor noise, sparsity of measurements, scan update rate, scan shape, and dependence from the motion noise of the vehicle. The same is true of the opposite, the same types of sensors will not behave the same in different environments because different materials do not reflect the emitted beams in the same way, and the presence of walls, open air, or moving objects greatly influence measurements.

In our research, an AUV is moving in a confined environment and uses an imaging sonar with a relatively slow mechanically rotating head. As explained in Chapter 3, the sparse measurements of the sonar are converted to a 360 deg scan when combined with the dead-reckoning information from the vehicle. So the question is: which scan matching algorithm should we use for our SLAM implementation?

To answer this question, we compared the major scan matching techniques with the real world Marina dataset (see Sec. 7.3.1) which provides ground truth. Each scan, after being formed, is matched with the previous one and the scan matching correction is applied to the dead-reckoning trajectory,

generating a corrected one. After the computation, the corrected trajectory is compared with the DGPS ground truth trajectory, resulting in a set of 217 comparisons for each algorithm. The sensor noise is quite high since it was recorded using a sonar sensor and because of the scan forming procedure: motion noise is propagated to the scan points. The motion noise is also high because the underwater instruments returned a dead-reckoning estimation that drifts over time. These characteristics make this dataset suitable for testing the quality of correction and the robustness to sensor noise.

To ensure fairness, all the algorithms were implemented in the same working environment under MATLAB and using information from the corresponding literature. During the implementation, no algorithmic optimization was used since the study focuses on the robustness and quality of correction. Each of the algorithms has at least a parameter that controls the general behavior and it has to be changed according to the scenario, scan size and error. To ensure that each algorithm worked at its best, research was performed for each algorithm to identify the optimal parameters for our dataset.

All the algorithms are the product of strong theoretical studies but not all are suitable for our implementation mostly due the sparseness of the scan points and their uncertainty which depends on the vehicle's motion. The mean and standard deviation of the module of the error for each tested algorithm across the full trajectory are summarized in Table 4.1. The general conclusion is that group-wise, the best behavior was given by the ICP-based algorithms for this type of correction problems: a robot scanning an underwater environment using a 2D sonar sensor. The other algorithms are valid but they cannot ensure a quality of correction comparable to the ICP-based methods with the exception of LF/SoG whose comparable to the other ICP-based methods.

The pIC proved to be the best among the algorithms. Its success should be attributed to its powerful handling of the uncertainties that permits the proper association of sparse points and results in better correction. For this reason, we selected the pIC algorithm which is applied analytically in the next chapter, together with our new algorithm for estimating its uncertainty.

Table 4.1: Mean and standard deviation of the module of error for each tested algorithm across the full trajectory

Methods	Algorithm	Mean (m)	Std (m)
ICP-style	ICP	0.7385	0.4412
	IDC	0.7490	0.4407
	MbICP	0.7366	0.4666
	pIC	0.7174	0.4302
Monte Carlo	VASCO	0.7945	0.4713
	DP-SLAM 2.0	0.8274	0.4939
	GA-ICP	0.7846	0.4576
Correlation	AH	1.1057	0.5527
	FMT	0.9874	0.5733
	HT	0.8264	0.4850
Newton's	NDT	1.0122	0.8739
	LF/SoG	0.7559	0.4638

Table 4.2: Summary of selected works on scan matching

	Authors	Algorithm name	Principal sensor	Dataset	Environment	Stochastic	Error estimation	Association method
ICP-style methods	Besl and McKay (1992)	ICP	Laser	sim/real	shapes	no	no	point-model
	Lu and Milios (1994)	IDC	Laser	sim/real	indoor	no	no	point-tangent/point
	Lu and Milios (1997b)	IDC	Laser	sim/real	indoor	yes	yes	point-tangent/point
	Chetverikov et al. (2002)	TrICP	Laser	sim	shapes	no	no	point-point
	Pfister et al. (2002)		Laser	real	indoor	yes	yes	point-line
	Bengtsson and Baerveldt (2003)	IDC-S	Laser	sim/real	indoor	yes	yes	point-line/point
	Zinßer et al. (2003)	Picky ICP	Laser	sim/real	shapes	no	no	point-point
	Mínguez et al. (2005)	MbICP	Laser	real	indoor	no	no	point-point
	Diosi and Kleeman (2005)	PSM	Laser	sim/real	indoor	yes	no	point-point
	Montesano et al. (2005)	pIC	Laser	real	indoor	yes	no	point-point
	Burguera et al. (2007)	spIC	Ultrasonic	real	indoor	yes	no	point-point
	Nieto et al. (2007)	ICP	Laser	sim/real	outdoor	yes	yes	point-model/point
	Censi (2007)	ICP	Rangefinder	sim	indoor	yes	yes	point-line
	Censi (2008)	PLICP	Laser	real	indoor	no	no	point-line
	Hernández et al. (2009)	MSISpIC	Imaging sonar	real	underwater	yes	no	point-point
Monte Carlo	Montemerlo et al. (2003)	VASCO	Laser	sim/real	indoor	yes	yes	grid-map
	Duckett (2003)		Laser	real	indoor	yes	yes	GA, grid-map
	Eliazar and Parr (2004)	DP-SLAM 2.0	Laser	real	indoor	yes	yes	PF, grid-map
	Martínez et al. (2006)	GA-ICP	Laser	real	outdoor	yes	yes	GA, point-point
	Ze-Su et al. (2007)	GPSM	Laser	sim/real	indoor	yes	yes	GA, point-point
	Lenac et al. (2007)	GLASM	Ultrasonic	real	indoor	yes	yes	GA, look-up table
	Zandara and Nicholson (2009)	SRUPF-GM	Laser	real	indoor	yes	yes	PF, grid map
Correlation	Weiß and von Puttkamer (1995)		Laser	real	indoor	yes	no	Histograms
	Censi et al. (2005)	HSM	Laser	sim	in/outdoor	yes	yes	Hough domain
	Bosse and Roberts (2007)		Laser	real	outdoor	yes	no	Histograms
	Bülow et al. (2010)	iFMT	Imaging sonar	real	underwater	yes	no	Spectral
	Pfingsthorn et al. (2010)	iFMI	Camera	real	underwater	yes	yes	Spectral
	Bülow and Birk (2011)	SRMR	Imaging sonar 3D	sim/real	underwater	yes	no	Spectral
	Pfingsthorn et al. (2012)		Imaging sonar 3D	sim/real	underwater	yes	yes	Spectral
	Hurtós et al. (2013)		Acoustic camera	real	underwater	yes	yes	Spectral
Newton's	Thrun (2001)		Laser	real	indoor	yes	yes	Likelihood fields
	Biber and Straßer (2003)	NDT	Laser	real	indoor	yes	yes	Probability densities
	Magnusson et al. (2007)	NDT	Laser	real	outdoor	yes	yes	Probability densities
	Burguera et al. (2009)	sNDT, LF/SoG	Ultrasonic	real	indoor	yes	yes	Likelihood fields

Chapter 5

Probabilistic Scan Matching

As was described in the previous chapter, several scan matching algorithms exist with most of them being variations of the ICP algorithm. The geometric representation of a scan in the conventional ICP algorithm does not model the uncertainty of the sensor measurements. Correspondences between two scans are chosen based on the closest-point rule normally using the Euclidean distance. As pointed out in Montesano et al. (2005), this distance does not take into account that the points in the new scan that are far from the sensor, could be far from their corresponding point in the previous scan due to the sensor uncertainty. In other words, if the scan data is very noisy, two statistically compatible points could appear far enough, in terms of their Euclidean distance. This situation may prevent a possible association or even generate a wrong one, so Montesano et al. (2005) proposed the pIC algorithm which is a statistical extension of the ICP algorithm in which the relative displacement and the observed points in both scans are modeled as GRVs. Because of the strong noise that characterizes underwater sonar data, we select the pIC algorithm for the scan matching part of our algorithm. To illustrate the modified pIC, it is provided in Algorithm 6 and explained below in detail.

The inputs of the algorithm are the reference scan S_{ref} with points $\mathbf{r}_i$ $(i = 1..n)$, the new scan S_{new} with points $\mathbf{c}_i$ $(i = 1..m)$ and the initial relative displacement estimation $\hat{\mathbf{q}}_0$ with its covariance $\mathbf{P}_q$ (line 1). The following

Algorithm 6 The modified pIC algorithm

1: $\hat{\mathbf{q}}_{pIC} = pIC(S_{ref}, S_{new}, \hat{\mathbf{q}}, \mathbf{P}_q)$ {
2: $k = 0$
3: $m = size(S_{new})$
4: $\hat{\mathbf{q}}_k = \hat{\mathbf{q}}$
5: **do** {
6: **for**$(i = 0; m; i++)$ {
7: $\hat{\mathbf{n}}_i = \hat{\mathbf{q}}_k \oplus \hat{\mathbf{c}}_i$
8: $A = \{\mathbf{r}_i \in S_{ref} / D_M^2(\mathbf{r}_i, \mathbf{n}_i) \leq \chi_{2,\alpha}^2\}$
9: $\hat{\mathbf{a}}_i = arg\ min_{\mathbf{r}_i \in A}\{D_M^2(\mathbf{r}_i, \mathbf{n}_i)\}$
10: $\hat{\mathbf{e}}_i = \hat{\mathbf{a}}_i - \hat{\mathbf{q}}_k \oplus \hat{\mathbf{c}}_i$
11: $\mathbf{P}_{e_i} = \mathbf{P}_{a_i} + \mathbf{J}_q \mathbf{P}_q \mathbf{J}_q^T + \mathbf{J}_c \mathbf{P}_{c_i} \mathbf{J}_c^T$
12: }
13: $\hat{\mathbf{q}}_{min} = arg\ min_{\mathbf{q}} \left\{ \frac{1}{2} \sum_{i=1}^{m} \left(\hat{\mathbf{e}}_i^T \mathbf{P}_{e_i}^{-1} \hat{\mathbf{e}}_i \right) \right\}$
14: **if**$(Convergence())$
15: $\hat{\mathbf{q}}_{pIC} = \hat{\mathbf{q}}_{min}$
16: **else** {
17: $\hat{\mathbf{q}}_{k+1} = \hat{\mathbf{q}}_{min}$
18: $k++$
19: }
20: }
21: **while**$(!Convergence()$ **and** $k < maxIterations)$
22: }

procedure is iteratively executed until convergence. First, the points of the new scan ($\mathbf{c}_i$) are compounded with the robot displacement ($\mathbf{q}_k$) (line 7). The result ($\mathbf{n}_i$), is the set of points from the new scan projected to the same reference frame as the S_{ref}. Then, for each point $\mathbf{n}_i$, the set of statistically compatible points (A) is computed using the *Mahalanobis* distance and a certain confidence level α (line 8). From that, the association point $\mathbf{a}_i$ is selected using the individual compatibility nearest neighbor (ICNN) criterion (Bar-Shalom and Fortmann, 1988) (line 9).

Given that $\mathbf{q}_k \equiv N(\hat{\mathbf{q}}_k, \mathbf{P}_q)$, $\mathbf{c}_i \equiv N(\hat{\mathbf{c}}_i, \mathbf{P}_{c_i})$ and $\mathbf{r}_i \equiv N(\hat{\mathbf{r}}_i, \mathbf{P}_{r_i})$ are GRVs, the matching error $\mathbf{e}_i \equiv N(\hat{\mathbf{e}}_i, \mathbf{P}_{e_i})$ of the $\{\mathbf{a}_i, \mathbf{n}_i\}$ pairing can be computed

as (lines 10-11):

$$\mathbf{e}_i = \mathbf{a}_i - \mathbf{n}_i = \mathbf{a}_i - \mathbf{q}_k \oplus \mathbf{c}_i \tag{5.1}$$

$$\hat{\mathbf{e}}_i \equiv \hat{\mathbf{a}}_i - \hat{\mathbf{q}}_k \oplus \hat{\mathbf{c}}_i \tag{5.2}$$

$$\mathbf{P}_{e_i} = \mathbf{P}_{a_i} + \mathbf{J}_q \mathbf{P}_q \mathbf{J}_q^T + \mathbf{J}_c \mathbf{P}_{c_i} \mathbf{J}_c^T \tag{5.3}$$

where

$$\mathbf{J}_q = \left.\frac{\partial\, \mathbf{a}_i - \mathbf{q}_k \oplus \mathbf{c}_i}{\partial\, \mathbf{q}_k}\right|_{\hat{\mathbf{q}}_k}, \qquad \mathbf{J}_c = \left.\frac{\partial\, \mathbf{a}_i - \mathbf{q}_k \oplus \mathbf{c}_i}{\partial\, \mathbf{c}_i}\right|_{\hat{\mathbf{c}}_i} \tag{5.4}$$

$\mathbf{P}_{e_i}$ is the uncertainty of the matching error $(\mathbf{a}_i - \mathbf{n}_i)$ used to estimate the displacement $\hat{\mathbf{q}}_{min}$ through the minimization of the squared *Mahalanobis* distance of the matching error (line 13). This is done using the non-linear least squares minimization method, iterating until convergence is achieved (lines 14-21).

5.1 Scan matching covariance estimation

Calculating the error covariance of a measurement is essential when it has to be combined with other measurements in a probabilistic framework like SLAM. For the Monte Carlo family of methods, this calculation comes at low computational cost because the particles' distribution can approximate the probability of the measurement. The same is true for Newton's methods because the scan matching error can be easily computed from the Hessian matrix used by Newton's algorithm.

For the correlation methods, the general idea relies on the fact that a perfect signal or image match results in a well-defined Dirac pulse. Hence, the amplitude and extent of values surrounding the main correlation peak can be used as a measure of the degree of congruence for the match. In Pfingsthorn et al. (2010), the result of the phase correlation is treated as a probability mass function and a covariance matrix is fitted to a neighborhood around the registration peak. The same idea has been later extended in Pfingsthorn et al. (2012) for the uncertainty estimation of a 6 DOF registration method used to register 3D sonar scans. Hurtós et al. (2013) proposed another heuristic

method by thresholding the correlation surface at half of the main peak amplitude and extracting the variances in both axes for the values exceeding the threshold.

5.1.1 pIC covariance estimation

Although ICP-style algorithms are very popular, lead to very good estimates of the relative displacements, and have been improved over the last decade, apparently very little research has attempted to address the uncertainty of their estimates. Covariance estimation based on the environment and the uncertainties of the sensor model has been introduced in Lu and Milios (1997b) and Pfister et al. (2002) but as has been shown in Bengtsson and Baerveldt (2003), this estimation can be very optimistic in a number of cases.

To overcome this problem, they suggest two different methods, the *Hessian* and the *sampling* methods. Because part of the scan matching algorithm defines the error function to be minimized, through the linearization of this error function, its Hessian matrix can be calculated and the covariance can be estimated by linear regression theory. The sampling method estimates the covariance matrix of a scan from a specific position by simulating and matching scans around the position. The authors conclude that the Hessian method is suitable for real-time estimation, with the ability to capture the shape but not the size of the covariance matrix. On the other hand, while the second method captures the correct size and shape of the covariance matrix, it can only be applied offline due to its computational cost.

In a SLAM framework, an approach similar to the offline method is used in Nieto et al. (2007), although in this case, only the initial guess of the displacement is sampled. In Balsamo et al. (2006), the authors suggested that the method for estimating the covariance should be independent from the algorithm used for the minimization. They observed the paradox that different optimization methods may converge to the same minimum, but lead to seemingly different uncertainties without indicating which one is to be adopted. Based on the same observation, the uncertainty propagation independent from the minimization algorithm for the computer vision domain

is shown in Haralick (1998). The method is based on the Hessian of the cost function with respect to the estimated displacement and the derivative of the Jacobian of the cost function with respect to the measurements. In Censi (2007), this method was adapted to ICP which propagates the uncertainty from the measured sonar points to the scan matching solution. Using Monte Carlo simulations over numerical implementation, the author reported similar results in corridors and circular scenarios and superior performance in square shaped environments than previous methods (Bengtsson and Baerveldt, 2003).

Hereafter, Haralick's method is adapted to our pIC algorithm for which a closed form solution is proposed.

Let $\mathbf{a}_i = [a_{x_i}, a_{y_i}]^T \in S_{ref}$ be the corresponding point of $\mathbf{c}_i = [c_{x_i}, c_{y_i}]^T \in S_{new}$ and $\mathbf{q}_k = [x, y, \theta]^T$ the predicted displacement. Then, the robot displacement $\mathbf{q}_{min} = [x, y, \theta]^T$ can be estimated through the minimization of the error function $\mathbf{e}_i = \mathbf{a}_i - \mathbf{q}_k \oplus \mathbf{c}_i$ (as described in Algorithm 6):

$$\begin{aligned} \hat{\mathbf{q}}_{min} &= arg\ min_q \left\{ f\left(S_{ref}, S_{new}, \mathbf{q}_k\right) \right\} = \\ &= arg\ min_q \left\{ \frac{1}{2} \sum_{i=1}^{n} \left(\hat{\mathbf{e}}_i^T \cdot \mathbf{P}_{e_i}^{-1} \cdot \hat{\mathbf{e}}_i \right) \right\} \end{aligned} \tag{5.5}$$

The method for propagating the uncertainties from the scan points to the scan displacement, independently from the minimization algorithm, only assumes that the cost function f has finite first and second partial derivatives with respect to sonar scan points $\mathbf{z} \equiv N\left(\hat{\mathbf{z}}, \mathbf{P}_z\right)$ and to the scan displacement $\mathbf{q}_k$. If we take the partial derivatives of f with respect to $\mathbf{q}_k$, we can form the gradient g:

$$g\left(\mathbf{z}, \mathbf{q}_k\right) = \frac{\partial f(\mathbf{z}, \mathbf{q}_k)}{\partial \mathbf{q}_k} \tag{5.6}$$

where

$$\hat{\mathbf{z}} = \left[a_{x_1}, a_{y_1}, n_{x_1}, n_{y_1} \cdots a_{x_k}, a_{y_k}, n_{x_k}, n_{y_k}\right]^T \tag{5.7}$$

By taking the Taylor series expansion of g around $(\hat{\mathbf{z}}, \hat{\mathbf{q}}_k)$ and taking into account that $\mathbf{q}_{min}$ is minimizing f and g, we can estimate the covariance $\mathbf{P}_q$

of the $\hat{\mathbf{q}}_{min}$ as:

$$\mathbf{P}_{q_{min}} = \left(\frac{\partial g}{\partial \mathbf{q}_{min}}\right)^{-1} \cdot \frac{\partial g}{\partial \mathbf{z}} \cdot \mathbf{P}_z \cdot \left(\frac{\partial g}{\partial \mathbf{z}}\right)^{T} \cdot \left(\frac{\partial g}{\partial \mathbf{q}_{min}}\right)^{-1} \tag{5.8}$$

where $\mathbf{P}_z$ is a matrix consisting of the uncertainties of the scan points.

The formulation of the covariance of (5.5) in closed-form is not trivial since the construction of the necessary matrices depends on the cost function and parameters. Next section, presents the closed-form expressions for calculating (5.8) in order to approximate the covariance of the scan matching estimation.

5.1.2 Closed-form formulation of the scan matching uncertainty

A closed-form formula for propagating the uncertainty from matched image pairs to homography parameters describing the image motion for a 2D image mosaic optimization problem, is presented in Elibol (2011). The formula is based on the first order approximation for the bundle adjustment (BA) minimization algorithm. Hereinafter, this method is adapted to the estimation of the covariance matrix of the scan matching displacement estimate. Let,

- $\mathbf{z} \equiv N(\hat{\mathbf{z}}, \mathbf{P}_z)$, be the vector of the measured scan points assumed to be perturbed with a zero mean Gaussian random noise. In our case, the scan matching does point-to-point association, therefore the measurements vector $\hat{\mathbf{z}}$ is of dimension $4k \times 1$:

$$\hat{\mathbf{z}} = [\underbrace{a_{x_1}, a_{y_1}, n_{x_1}, n_{y_1}}_{\hat{\mathbf{z}}_1} \cdots \underbrace{a_{x_k}, a_{y_k}, n_{x_k}, n_{y_k}}_{\hat{\mathbf{z}}_k}]^T \tag{5.9}$$

  and its covariance, given by $\mathbf{P}_z$, is a $(4k \times 4k)$ matrix consisting of the

uncertainties of the scan points:

$$\mathbf{P}_z = \begin{bmatrix} \mathbf{P}_{a_1} & & \cdots & & 0 \\ & \mathbf{P}_{n_1} & & & \\ \vdots & & \ddots & & \vdots \\ & & & \mathbf{P}_{a_k} & \\ 0 & & \cdots & & \mathbf{P}_{n_k} \end{bmatrix} \tag{5.10}$$

which is block diagonal because the scan points are assumed to be uncorrelated. One of the differences with laser scanners or multi-beam sonar profilers with the ones using a rotating mono-beam sonar head, is that the scan points become correlated when represented in the scan I frame. Nevertheless, given the short duration of the scan building process, the slow motion of the vehicle and without loss of generality, those correlations have been neglected in this research effort. Let,

- $\mathbf{x}$ be the unknown parameter vector corresponding to the $\mathbf{q}_{min}$ estimated by the pIC.

$$\mathbf{x} = [x, y, \psi]^T \tag{5.11}$$

- $f(\mathbf{z}, \mathbf{x})$ be a scalar, continuous, non-negative cost function of the overall square Mahalanobis distance of the matching error (from algorithm 1, line 13):

$$f(\mathbf{z}, \mathbf{x}) = \frac{1}{2} \sum_{i=1}^{k} \left(\mathbf{e}_i^T \cdot \mathbf{P}_{e_i}^{-1} \cdot \mathbf{e}_i \right) \tag{5.12}$$

Haralick's method (Haralick, 1998) can then be applied to estimate the covariance $\mathbf{P}_x$ of the estimated $\hat{\mathbf{x}}$ which minimizes the above cost function (5.12):

$$\mathbf{P}_x = \left(\frac{\partial g}{\partial \mathbf{x}} \right)^{-1} \cdot \frac{\partial g}{\partial \mathbf{z}} \cdot \mathbf{P}_z \cdot \left(\frac{\partial g}{\partial \mathbf{z}} \right)^T \cdot \left(\frac{\partial g}{\partial \mathbf{x}} \right)^{-1} \tag{5.13}$$

where $g(\mathbf{z}, \mathbf{x}) = \left[\frac{\partial f(\mathbf{z}, \mathbf{x})}{\partial \mathbf{x}} \right]^T$. To do this, it is necessary to compute $g(\mathbf{z}, \mathbf{x})$, $\frac{\partial g(\mathbf{z}, \mathbf{x})}{\partial \mathbf{x}}$ and $\frac{\partial g(\mathbf{z}, \mathbf{x})}{\partial \mathbf{z}}$.

Let us begin by rewriting the function (5.12) as:

$$f(\mathbf{z}, \mathbf{x}) = \frac{1}{2} \cdot \mathbf{R}^T \cdot \mathbf{W} \cdot \mathbf{R} \tag{5.14}$$

where $\mathbf{R}$ is the stacked vector of the matching residuals error (of dimension $2k \times 1$):

$$\mathbf{R} = \left[\mathbf{e}_1^T \cdots \mathbf{e}_k^T\right]^T \tag{5.15}$$

and $\mathbf{W}$ is the inverted block diagonal matrix of the measurement errors' covariances (of dimension $2k \times 2k$):

$$\mathbf{W} = \begin{bmatrix} \mathbf{P}_{e_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{P}_{e_k} \end{bmatrix}^{-1} \tag{5.16}$$

Because $\mathbf{W}$ is the inverse of a covariance matrix, it is positive definite and hence a Cholesky decomposition exists:

$$\mathbf{W} = \mathbf{L}^T \cdot \mathbf{L} \tag{5.17}$$

Now, for simplicity and without loss of generality, let us define:

$$\hat{\mathbf{R}} = \mathbf{L} \cdot \mathbf{R} \tag{5.18}$$

$\mathbf{W}$ is block diagonal because the scan points are assumed to be uncorrelated, and so is $\mathbf{L}$:

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_{e_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{L}_{e_k} \end{bmatrix} \tag{5.19}$$

Now, Equation 5.14 can be rewritten as:

$$f(\mathbf{z}, \mathbf{x}) = \frac{1}{2} \cdot \hat{\mathbf{R}}^T \cdot \hat{\mathbf{R}} \tag{5.20}$$

and $g(\mathbf{z}, \mathbf{x})$ can be defined as the Jacobian of the cost function, being a (1×3) matrix;

$$g\left(\mathbf{z}, \mathbf{x}\right) = \frac{\partial f}{\partial x} = \hat{\mathbf{R}}^T \cdot \hat{\mathbf{J}}_x \tag{5.21}$$

where $\hat{\mathbf{J}}_x$ is the $(2k \times 3)$ Jacobian matrix of error vector $\hat{\mathbf{R}}$:

$$\hat{\mathbf{J}}_x = \frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial(\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial x} & \frac{\partial(\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial y} & \frac{\partial(\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial \theta} \\ \vdots & \vdots & \vdots \\ \frac{\partial(\mathbf{L}_{e_k} \cdot \mathbf{e}_k)}{\partial x} & \frac{\partial(\mathbf{L}_{e_k} \cdot \mathbf{e}_k)}{\partial y} & \frac{\partial(\mathbf{L}_{e_k} \cdot \mathbf{e}_k)}{\partial \theta} \end{bmatrix} \tag{5.22}$$

The $\frac{\partial g(\mathbf{z},\mathbf{x})}{\partial \mathbf{x}}$ is the (3×3) Hessian of $f\left(\mathbf{z}, \mathbf{x}\right)$ and is calculated as follows:

$$\frac{\partial g}{\partial \mathbf{x}} = 2 \cdot \hat{\mathbf{J}}_x^T \cdot \hat{\mathbf{J}}_x + 2 \cdot \hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{x}} \tag{5.23}$$

where $\frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{x}}$ is the $(6k \times 3)$ Hessian of $\hat{\mathbf{R}}$ which can be computed in the following way:

$$\frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}} \left(\frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} \right) = \sum_{i=1}^{3} \left(vec \left(\frac{\partial \hat{\mathbf{J}}_x}{\partial x_i} \right) \right) \cdot \mathbf{r}_i^T \tag{5.24}$$

being $\mathbf{r}_i$ a (3×1) vector, with all zeros except its i^{th} row which is equal to 1. To compute the second part of (5.23), the (5.24) is multiplied by $\hat{\mathbf{R}}^T$ as follows:

$$\hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{x}} = \left(\hat{\mathbf{R}}^T \otimes \mathbf{I}_3 \right) \cdot \frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{x}} \tag{5.25}$$

where $\otimes$ denotes Kronecker product of two matrices. Similarly, the $\frac{\partial g}{\partial \mathbf{z}}$ is a $3 \times 4k$ matrix, computed as:

$$\frac{\partial g}{\partial \mathbf{z}} = 2 \cdot \hat{\mathbf{J}}_x^T \cdot \hat{\mathbf{J}}_z + 2 \cdot \hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{z}} \tag{5.26}$$

where $\hat{\mathbf{J}}_z$ is the Jacobian of the error vector $\hat{\mathbf{R}}$, being a $(2k \times 4k)$ matrix:

$$\hat{\mathbf{J}}_z = \frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{z}} = \begin{bmatrix} \frac{\partial \hat{\mathbf{R}}_1}{\partial \mathbf{z}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\partial \hat{\mathbf{R}}_k}{\partial \mathbf{z}} \end{bmatrix} \tag{5.27}$$

where each $\frac{\partial \hat{\mathbf{R}}(i)}{\partial \mathbf{z}}$ is a (2×4) matrix, and $\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{z}}$ is the following $(6k \times 4k)$ matrix:

$$\frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{z}} = \sum_{i=1}^{4k} \left(vec\left(\frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{z}_i} \right) \right) \cdot \mathbf{r}_i^T \tag{5.28}$$

being this time r_i a $(4k \times 1)$ vector of all zeros except its i^{th} row which is equal to 1. As previously, the second part of (5.26) is given by:

$$\hat{\mathbf{R}}^T \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} \right) = \left(\hat{\mathbf{R}}^T \otimes \mathbf{I}_3 \right) \cdot \frac{\partial \hat{\mathbf{J}}_x}{\partial \mathbf{z}} \tag{5.29}$$

5.2 Validation of the covariance estimation

5.2.1 Numerical estimation

In order to verify the above closed-form formulation, we ran extensive tests comparing our results with a numerical estimation of the covariance. In the source code publicly available by Censi (2007), a numerical estimation of the covariance matrix based on (5.13), is used. We adapted this method to our case, as it provides an independent and tested algorithm for validation of our closed-form expression. The results were exactly the same as in the closed-form, rounded up to the fifth significant digit. The numerical method only needs to express the cost function (5.5) in an analytical way that includes all the components which contribute to the error.

$$\mathbf{E}(a_i, n_i, \hat{\mathbf{q}}_k) = \frac{1}{2} \sum \left[\mathbf{a}_i - \hat{\mathbf{q}}_k \oplus \mathbf{n}_i \right]^T \cdot \mathbf{P}_{e_i}^{-1} \cdot \left[\mathbf{a}_i - \hat{\mathbf{q}}_k \oplus \mathbf{n}_i \right] \tag{5.30}$$

Using (5.30), all the matrices needed to evaluate (5.13) are computed and then evaluated.

5.2.2 Monte-Carlo simulations

In order to compare the accuracy of the estimated covariance matrix given by (5.13) with the real one, we performed Monte-Carlo simulations. In this experiment, we consider two environments that are most common in

underwater man-made structures (Ribas et al., 2008); square and corridor.

As the sensor of our interest is an acoustic underwater sonar, the simulation constants were kept as close to reality as possible. The sensor has modeled with 200 acoustic beams distributed on a full sector of 360 deg, perturbed by Gaussian noise with zero mean and a standard deviation of 0.05 m for the range and 1.5 deg for the angle. The initial pose is $\mathbf{x}_{ref} = [0m, 0m, 0^o]^T$ and the true robot displacement is $\mathbf{q} = [1m, 0.5m, 10^o]^T$. The covariance of the dead-reckoning estimate $\mathbf{u}$, used as the initial guess for the pIC with $\mathbf{\Sigma}_u = diag(0.35m, 0.35m, 7.5^o)$.

The algorithm of the simulation followed the method proposed by Censi (2007):

1. Compute the second pose as $\mathbf{x}_{new} = \mathbf{x}_{ref} \oplus \mathbf{q}$
2. For $i = 1$ to 300 do
   (a) Create a noise-corrupted first scan S_{ref_i} at $\mathbf{x}_{ref}$
   (b) Create a noise-corrupted second scan S_{new_i} at $\mathbf{x}_{new}$
   (c) Sample a guess displacement $\mathbf{u}_i \sim N(\mathbf{q}, \mathbf{\Sigma}_u)$
   (d) Run pIC to obtain $\hat{\mathbf{q}}_i = pIC(S_{ref_i}, S_{new_i}, \mathbf{u}_i)$
3. Compute the approximated $\mathbf{\Sigma}_q$ as the covariance of the resulting samples $\hat{\mathbf{q}}_i$
4. Compare the Monte Carlo approximation $\mathbf{\Sigma}_q$ with the $\mathbf{\Sigma}_x$ estimated by (5.13)

Square environment

In Fig. 5.1 the top image and the results of the square environment, are visualized. Although difficult to appreciate in the figure, the dots in the wall are the points from the two scans (real and observed). The figure shows in red the covariance estimated from the samples of the Monte Carlo simulation and in blue the one estimated by the proposed method. It is possible to see that

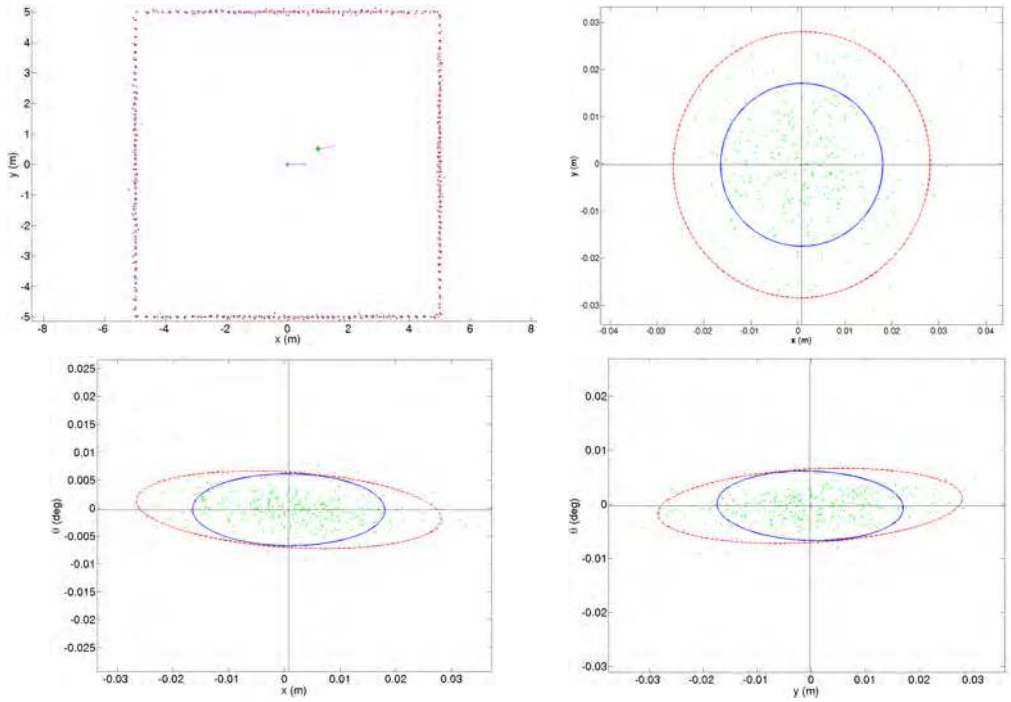


Figure 5.1: Square environment. The arrows on the top image represent the pose and forward direction of the robot. The error distribution on the second pose (right) is too small to be visualized. In the data projection plots, the dots (green) are the actual error samples. The computed covariance is represented with solid line (blue), while the real one is represented with a dashed line (red).

the method is slightly optimistic and does not capture well the correlation among x, y and θ. Although not shown in the figure, it is worth noting that the covariance estimated by the proposed closed form is in agreement with the one derived numerically with the method proposed in Censi (2007).

Corridor environment

The map and the results in the corridor environment are showing in Fig. 5.2. Although this is a very difficult environment for a scan matching algorithm, the proposed method captures well the real covariance matrix in the observed manifold (x degree of freedom). In this case the method is able to capture even the $x - \theta$ correlation. Table 5.1, shows the standard deviation results

for both environments.

Table 5.1: Monte Carlo standard deviation errors

Environment		$\sigma_x(mm)$	$\sigma_y(mm)$	$\sigma_\theta(mm)$
Square	true	11.2	11.5	0.16
	result	7.0	7.1	0.15
Corridor	true	9.8	257.2	0.22
	result	8.2	13.8	0.22

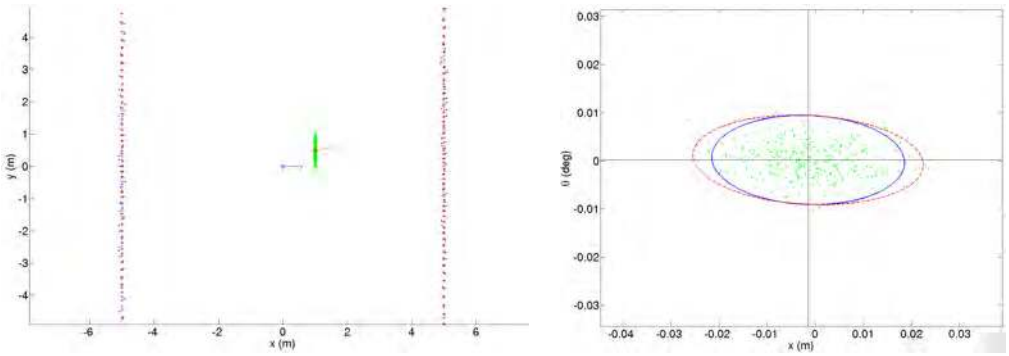


Figure 5.2: Corridor environment. Here is more easy to appreciate on the map the error distribution in the direction of the unobserved walls. Again, the arrows on the map represent the pose and forward heading of the robot. In the data projection plot, the dots (green) are the actual error samples. The solid line (blue) represents the computed covariance, while the dashed line (red) represents the real one.

5.3 Discussion

The scan matching covariance estimation is an important element for smooth integration to any probabilistic-based estimator. The closed-form solution that was presented in this Chapter, is based on Haralick's method (Equation 5.8), where $\mathbf{P}_z$ is a matrix consisting of the uncertainties of the scan points that are propagated to form the uncertainty of the scan matching estimation. The

experimental validation shows that the closed form solution provides a slightly under-estimated covariance when compared to the empirical estimate. We believe that this is due the current construction of the $\mathbf{P}_z$ matrix which is built as a block diagonal, not including the correlations between the scan points. Theoretically, $\mathbf{P}_z$ can be a full matrix describing the correlations between the scan points, however, practically estimating these correlations is not always trivial.

Under-estimated scan matching covariance, when used within a SLAM framework, can increase fault-positive registrations and loop-closures, affecting the overall trajectory estimation. Further mathematical research will help to include the correlations into the matrix in a consistent way, which potentially would improve the scan matching covariance estimation.

Chapter 6

Underwater SLAM algorithm

The proposed pose-based SLAM algorithm uses an ASEKF for estimating scan poses. In this implementation of the stochastic map (Smith et al., 1990), the estimate of the positions of the vehicle at the center of each full scan $\{\hat{\mathbf{x}}_1^B \dots \hat{\mathbf{x}}_n^B\}$ at time step (k) are stored in the state vector $\hat{\mathbf{x}}_k^B$, referenced to the base frame B:

$$\hat{\mathbf{x}}_k^B = \left[\left[\hat{\mathbf{x}}_{n_k}^B\right]^T \ \dots \ \left[\hat{\mathbf{x}}_{i_k}^B\right]^T \ \dots \ \left[\hat{\mathbf{x}}_{1_k}^B\right]^T \right]^T \tag{6.1}$$

The covariance matrix for this state is defined as:

$$\mathbf{P}_k^B = E\left[(\mathbf{x}_k^B - \hat{\mathbf{x}}_k^B)(\mathbf{x}_k^B - \hat{\mathbf{x}}_k^B)^T\right] \tag{6.2}$$

For the reader's convenience, we would like to reiterate that a full scan is defined as the set of range measurements obtained after compounding all the robot poses from the 200 beams needed to obtain the full 360 deg sonar scan sector. As mentioned in Chapter 3, the output from the *ScanGrabbing* algorithm is a scan referenced at the center of that path. Each scan is considered rigid and is statistically independent as it has been build on a local frame. For a better understanding, the SLAM is detailed in Algorithm 7 and explained afterward in detail.

Algorithm 7 SLAM algorithm

$SLAM()$ {
  {-Initialization}
  $[S_{ref}, \hat{\mathbf{q}}^{B_1}_{ref}, \mathbf{P}^{B_1}_{q_{ref}}] = ScanGrabbing()$
  $\mathbf{x}^B_1 = \hat{\mathbf{q}}^{B_1}_{ref},\ \mathbf{P}^B_1 = \mathbf{P}^{B_1}_{ref},\ S_1 = S_{ref}$

  {-Main}
  for $k = 2$ **to** n_scans **do**
  {
    $[S_{new}, \hat{\mathbf{q}}^{B_i}_{new}, \mathbf{P}^{B_i}_{q_{new}}] = ScanGrabbing()$
    $S_k = S_{k-1} \cup S_{new}$

    {-EKF prediction}
    $\mathbf{x}^B_k = \mathbf{x}^B_k \odot \hat{\mathbf{q}}^{B_i}_{new}$
    $\mathbf{P}^B_k = \mathbf{F}_k \mathbf{P}^B_k \mathbf{F}^T_k + \mathbf{G}_k \mathbf{P}^{B_i}_{q_{new}} \mathbf{G}^T_k$

    {-Loop candidates}
    $[\mathbf{O}_k] = OverlappingScans(S_k, \mathbf{x}^B_k, threshold)$
    for *all* $[S_i, \mathbf{x}^B_i] \in \mathbf{O}_k$ **do**
    {
      $S_{ref} = S_i$ $(ascending\ order)$
      $\mathbf{H}_k = [\mathbf{J}_{2\oplus_{3x3}}\ \mathbf{0}_{3x3(n-i-1)}\ \mathbf{J}_{1\oplus}\mathbf{J}_{\ominus_{3x3}}\ \mathbf{0}_{3x3(i-1)}]$
      $[\hat{\mathbf{q}}^{I_i}_i, \mathbf{P}^{I_i}_i] = [\ominus \hat{\mathbf{x}}^B_{i_k} \oplus \hat{\mathbf{x}}^B_{n_k}, \mathbf{H}_k \mathbf{P}^B_k \mathbf{H}^T_k]$
      $[\hat{\mathbf{q}}^{I_i}_{pIC}, \mathbf{P}^{I_i}_{pIC}] = pIC(S_{ref}, S_{new}, \hat{\mathbf{q}}^{I_i}_i, \mathbf{P}^{I_i}_i)$

      {-EKF update}
      $z_k = \hat{\mathbf{q}}^{I_i}_{pIC};\ R_k = \mathbf{P}^{I_i}_{pIC}$
      $[\mathbf{x}^B_k, \mathbf{P}^B_k] = standardEKFupdate(\mathbf{x}^B_{k|i}, \mathbf{P}^B_{k|i}, z_k, R_k, \mathbf{H}_k)$
    }
  }
}

6.1 Map initialization

All the elements of the state vector are represented in the map reference frame B. Although this reference frame can be defined arbitrarily, we have chosen to place its origin at the initial position of the vehicle at the beginning of the experiment and to orient it to the North so compass measurements can

be easily integrated.

The pose state $\mathbf{x}_i$ is represented as:

$$\mathbf{x}_i^B = [x \ y \ \psi]^T \tag{6.3}$$

where x, y and ψ are the position and orientation of the vehicle in the global frame B. The depth is not considered in this case as we are developing a planar map of the environment. The state and the map are initialized from the first available heading measurement.

6.2 Prediction

Let,

- $\mathbf{x}_{n-1_k}^B \equiv N(\hat{\mathbf{x}}_{n-1_k}^B, \ \mathbf{P}_k^B)$ be the last scan pose, and
- $\mathbf{q}_n^{B_n} \equiv N(\hat{\mathbf{q}}_n^{B_n}, \ \mathbf{P}_{q_n}^{B_n})$ be the robot displacement during the last scan, estimated through dead-reckoning.

Then the prediction / state augmentation equation is given by:

$$\hat{\mathbf{x}}_{k+1}^{B^+} = \left[\left[\hat{\mathbf{x}}_{n-1_k}^B \odot \hat{\mathbf{q}}_{n_k}^{B_n} \right]^T \left[\hat{\mathbf{x}}_{n-1_k}^B \right]^T \dots \left[\hat{\mathbf{x}}_{i_k}^B \right]^T \dots \left[\hat{\mathbf{x}}_{1_k}^B \right]^T \right]^T \tag{6.4}$$

where, given that B and B_n frames are both North aligned, the linear operator $\odot$ is defined in the general case as:

$$\mathbf{x} \odot \mathbf{q} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \odot \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \begin{bmatrix} a+d \\ b+e \\ f \end{bmatrix} \tag{6.5}$$

Let $\mathbf{A}_\odot$ and $\mathbf{B}_\odot$ be the corresponding linear transformation matrices defined as:

$$\mathbf{A}_\odot = \begin{bmatrix} \mathbf{I}_{2x2} & \mathbf{0}_{2x1} \\ \mathbf{0}_{1x2} & 0 \end{bmatrix}, \quad \mathbf{B}_\odot = \mathbf{I}_{3x3} \tag{6.6}$$

and let $\mathbf{P}_{k+1}^{B+}$ be the predicted pose uncertainty computed as:

$$\mathbf{P}_{k+1}^{B+} = \mathbf{F}_k \mathbf{P}_k^B \mathbf{F}_k^T + \mathbf{G}_k \mathbf{P}_{q_n}^B \mathbf{G}_k^T \tag{6.7}$$

where,

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{A}_{\odot} & \mathbf{0}_{3x3} & \dots & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x3} & \mathbf{I}_{3x3} & \dots & \mathbf{0}_{3x3} \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{0}_{3x3} & \mathbf{0}_{3x3} & \dots & \mathbf{I}_{3x3} \end{bmatrix}, \quad \mathbf{G}_k = \begin{bmatrix} \mathbf{B}_{\odot} \\ \mathbf{0}_{3x3} \\ \vdots \\ \mathbf{0}_{3x3} \end{bmatrix} \tag{6.8}$$

6.3 Scan matching measurement

In order to execute the modified pIC algorithm given two overlapping scans (S_i, S_n) with their related poses $(\mathbf{x}_i^B, \mathbf{x}_n^B)$, an initial guess of their relative displacement is necessary. This initial guess $[\hat{\mathbf{q}}_i^{I_i}, \mathbf{P}_{q_i}^{I_i}]$ can be easily extracted from the state vector using the tail-to-tail transformation (Smith et al., 1990):

$$\hat{\mathbf{q}}_i^{I_i} = \ominus \hat{\mathbf{x}}_i^B \oplus \hat{\mathbf{x}}_n^B \tag{6.9}$$

Since the tail-to-tail transformation is actually a non-linear function of the state vector $\mathbf{x}_k^B$, the uncertainty of the initial guess can be computed by means of the Jacobian of the non-linear function:

$$\mathbf{P}_{q_i}^{I_i} = \mathbf{H}_k \mathbf{P}_k^B \mathbf{H}_k^T \tag{6.10}$$

where

$$\mathbf{H}_k = \left. \frac{\partial \ \ominus \mathbf{x}_i^B \oplus \mathbf{x}_n^B}{\partial \ \mathbf{x}_k^B} \right|_{\hat{\mathbf{x}}_k^B} \tag{6.11}$$

Moreover, as shown in Smith et al. (1990), the Jacobian for the tail-to-tail transformation $\mathbf{x}_{a_c} = \ominus \mathbf{x}_{b_a} \oplus \mathbf{x}_{b_c}$, is:

$$\frac{\partial \ \ominus \mathbf{x}_{b_a} \oplus \mathbf{x}_{b_c}}{\partial \ (\mathbf{x}_{b_a} \mathbf{x}_{b_c})} = [\mathbf{J}_{1\oplus} \mathbf{J}_{\ominus} \quad \mathbf{J}_{2\oplus}] \tag{6.12}$$

where $\mathbf{J}_{1\oplus}, \mathbf{J}_{2\oplus}$ and $\mathbf{J}_{\ominus}$ are the Jacobian matrices of the compounding and inverse transformations respectively.

As in our case $\mathbf{x}_{n_k}^B$ and $\mathbf{x}_{i_k}^B$ are components of the full state vector, the Jacobian of the measurement equation becomes:

$$\begin{aligned}\mathbf{H}_k =& \frac{\partial \ \ominus \mathbf{x}_{i_k}^B \oplus \mathbf{x}_{n_k}^B}{\partial \, \mathbf{x}_k} = \\ =& [\mathbf{J}_{2\oplus_{3x3}} \quad \mathbf{0}_{3x3(n-i-1)} \quad \mathbf{J}_{1\oplus}\mathbf{J}_{\ominus_{3x3}} \quad \mathbf{0}_{3x3(i-1)}]\end{aligned} \tag{6.13}$$

Once the initial displacement guess is available, the modified pIC algorithm can be used to produce an updated measurement of this displacement.

6.4 Loop closing candidates

Each new pose of a scan is compared against all the previous scan poses that are in the nearby area (defined by a threshold) using scan matching. In order to avoid false positive registrations we define a percentage between the number of points of the new scan and the associated ones from the candidates that take part in the registration . The scan matching result is accepted if enough points from the scans overlap (over the percentage), and the registration introduces a constraint between the poses, hence updating the ASEKF. These constraints are the loops closures that correct the whole trajectory and bound the drift.

Wrong matches in loop closing can have catastrophic results in any SLAM framework. To avoid that, we set the percentage of associated scan points up to 80 %, which means that should be a good overlapping between the scans in order the match to be accepted. If the match is not accepted, then the filter continues with the dead-reckoning until the next match. The same is true with occlusions. Even if the vehicle is in the exact point of a previous scan but an occlusion prohibit the same view, then there will not be enough associated points for a match and the filter will continue with the dead-reckoning.

Bellow follows the loop closure procedure. Let,

- $\mathbf{x}_{n_k}^B$ be the last scan pose and S_{n_k} its corresponding scan,
- $Overlap_k = \left\{ S_{i_k} \ / \ \left\| \hat{\mathbf{x}}_{n_k}^B - \hat{\mathbf{x}}_{i_k}^B \right\| < threshold \right\}$ the set of overlapping scans and

- $\mathbf{O}_k = [S_{1_k},\ S_{2_k}\ \ldots\ S_{m_k}]$ the sequence of the overlapping scans belonging to the $Overlap_k$ set

Then $\forall\ [S_{i_k}, \mathbf{x}_i^B] \in \mathbf{O}_k$, perform a new scan matching between the scan poses $(\mathbf{x}_{n_k}^B, \mathbf{x}_{i_k}^B)$ with the corresponding scans (S_{n_k}, S_{i_k}) obtaining $[\hat{\mathbf{q}}_i^{I_i}, \mathbf{P}_{q_i}^{I_i}]$ as the result of the scan matching. $\mathbf{P}_{q_i}^{I_i}$ is the corresponding uncertainty computed as described in Section 5.1. Finally the scan matching result is used to update the filter.

6.5 State update

When two overlapping scans (S_i, S_n) with the corresponding poses $(\mathbf{x}_i^B, \mathbf{x}_n^B)$ are registered, their relative displacement defines a constraint between both poses. This constraint can be expressed by means of the measurement equation, which in our case becomes:

$$\mathbf{z}_k = \ominus \mathbf{x}_{i_k}^B \oplus \mathbf{x}_{n_k}^B \tag{6.14}$$

where $\mathbf{x}_{i_k}^B$ is the scan pose which overlaps with the last scan pose $\mathbf{x}_{n_k}^B$. Now, an update of the stochastic map can be performed with the standard extended Kalman filter equations.

Chapter 7

Experimental set-up and results

The validity of the scan matching SLAM technique that was described in the previous sections, has been tested with three datasets. The first is obtained by the *Ictineu* AUV in an abandoned marina, the second in a natural underwater tunnel and the third in an underwater cavern system. Both the second and the third datasets were obtained using the *Sparus* AUV. The *Ictineu* and *Sparus* AUVs were designed and developed in the underwater robotics lab (CIRS) of the Universitat de Girona (UdG), Spain.

7.1 Ictineu AUV

The *Ictineu* AUV was conceived around a typical open-frame design (see Fig. 7.1a) as a research prototype for validating new technologies. It is a small ($0.8 \times 0.5 \times 0.5\,\mathrm{m}$) and light weight (60 kg in air) vehicle appropriate for very shallow waters (max depth 30 m). Although the hydrodynamics of open-frame vehicles is known to be less efficient than that of closed-hull type vehicles, they allow easy upgrades, sensor integration, and maintenance. The vehicle is passively stable in roll and pitch due to the weight and volume distribution, and it can be controlled in surge, heave, and yaw with four thrusters (two for the surge and yaw, and two for the heave DOF). Two cylindrical pressure vessels house the power and computer modules. The power module contains a pack of sealed lead acid batteries, which supply

24 Ah at 24 V and can provide the *Ictineu* AUV with more than one hour of running time. The computer module has two PCs, one for control (PC104 AMD GEODE-300 MHz) and one for image and sonar processing (mini-ITX computer Via C3 1 GHz) connected through a 100 Mbps Ethernet switch. Table 7.1 reports the sensor suite of the vehicle used when the dataset was collected. A complete description of the *Ictineu* AUV at that time can be found in Ribas et al. (2007).

7.2 Sparus AUV

The *Sparus* AUV, was designed with the main goal of being a small and simple torpedo-shaped vehicle with hovering capabilities (Fig. 7.1b). It has three DOF and the propulsion consists of three thrusters (two for the surge and yaw and one for the heave DOF) which are integrated in a classical torpedo shape AUV. The dimensions of the vehicle are 1.22 m in length by 0.23 m diameter, with a weight of around 30 kg and a depth rating of 50 m. The vehicle's power module consists of Lithium ion battery cells (890 Wh), which allows for an autonomy of more than 6 hours. The on-board embedded computer has been chosen as a trade-off between processing power, size and power consumption. The ADL945HD board together with the U2500 processor at 1.2 GHz provide the processing power of a Core Duo architecture together with the ultra low voltage consumption and the 3.5" small form factor.

The vehicle's sensor suite is listed in Table 7.1. Additional temperature, voltage, pressure sensors, and water leak detectors are installed into the pressure vessels to monitor the system's integrity. Besides, the vehicle hosts on top a WiFi and a GPS antenna covered with resin, which can be detached and placed in a float on the surface, to maintain the connection with the AUV via a 5 m USB cable while the vehicle is submerged.

Table 7.1: Summary of the Ictineu and Sparus AUVs sensor suit. (*Installed additionally for the cavern dataset)

Sensor	Ictineu AUV Model / Specifications	Sparus AUV Model / Specifications
MS Imaging sonar	Tritech Miniking	Tritech Micron DST
Max range:	100 m	75 m
Horizontal beamwidth:	3 deg	3 deg
Vertical beamwidth:	40 deg	35 deg
Scan rate (360-deg sector):	5-20 sec	5-20 sec
Frequency:	675 kHz	Chirped 650 to 750 kHz
MS Profiling sonar*	–	Tritech Super SeaKing DFP
Frequency:	–	0.6 / 1.1 MHz
Max range:	–	80 / 40 m
Beamwidth:	–	2 / 1 deg
Scan rate (360-deg sector):	–	4-25 sec
Doppler velocity log	Sontek Argonaut	LinkQuest NavQuest600Micro
Accuracy:	0.2%	0.2% +/- 1 mm/s
Frequency:	1.5 MHz	600 kHz
Max update rate:	10 Hz	5 Hz
AHRS 1	Xsens MTi	
Static accuracy (roll/pitch):	0.5 deg	
Static accuracy (heading):	1.0 deg	
Max update rate:	256 Hz	
AHRS 2*	–	Analog Devices ADIS16480
Static accuracy (roll/pitch):	–	0.1 deg
Static accuracy (heading):	–	0.3 deg
Max update rate:	–	330 Hz
Vision system 1	cameras (1 × forward and 1 × down-looking)	
CCD size:	1/3 inch	
Sensitivity:	0.01 lux	
Color encoding system:	PAL 576i	
Vision system 2*	–	GoPro Hero2 3D
CCD size:	–	1/2.5 inch
Image format:	–	1280 x 960
Field of view:	–	170 deg
Pressure sensor	Integrated to DVL	DS2806 HPS-A
Range:	0 - 20 bar	0 - 6 bar
Precision:	n.a.	+/- 0.25%

a) b)

Figure 7.1: The AUVs from the University of Girona that were used for this research. a) The *Ictineu* AUV and b) the *Sparus* AUV.

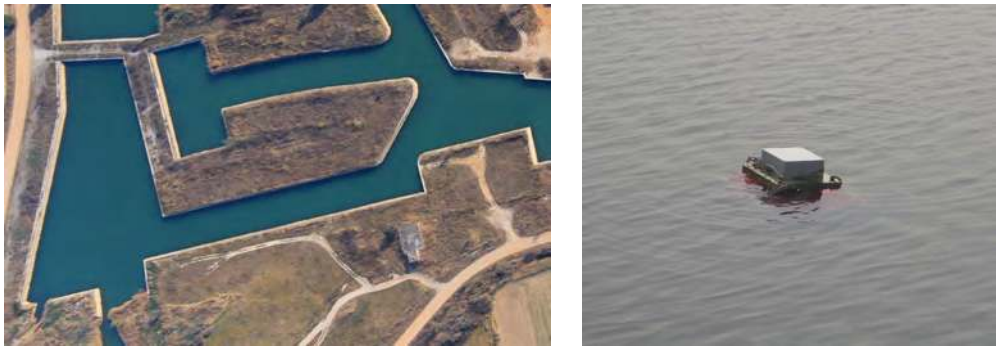
7.3 The datasets

The proposed method has been tested with three datasets, including a structured environment of an abandoned marina and the unstructured environments of a natural underwater tunnel and a natural underwater cave, all acquired during engineering tests along the Spanish coast.

7.3.1 The marina dataset

The method described in this paper has been first tested with a dataset obtained in an abandoned marina located at St. Pere Pescador, Costa Brava, Spain (Fig. 7.2a). This available dataset (Ribas et al., 2008) is useful to test if an algorithm is capable of utilizing the limited information provided by each scan in a large underwater environment. The dataset was obtained in a structured environment, however, our algorithm does not take into account any structural information or take advantage of existing features. In our current sensor configuration with an imaging sonar mechanically scanning the horizontal plane around the vehicle, the proposed method can be used wherever the surrounding vertical obstacles produce constant cross sections for the commanded operating depth.

The survey mission was carried out using the *Ictineu* AUV traveling along a 600 m path. The MSIS was configured to scan the whole 360 deg sector



a) b)

Figure 7.2: Gathering the marina dataset: a) the marina environment. b) The buoy on top of the AUV.

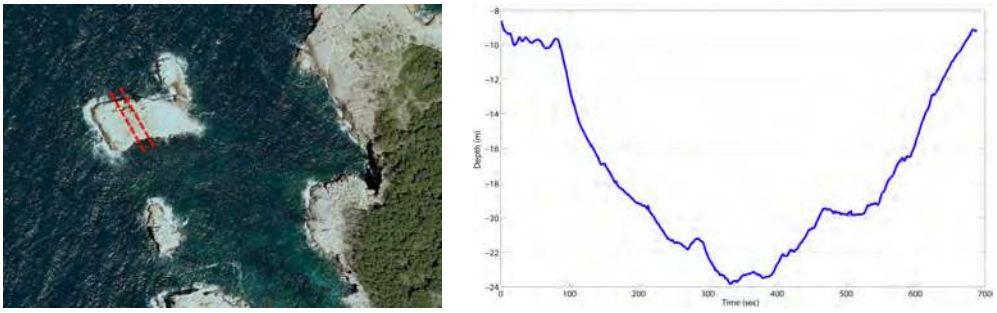
at a maximum range of 50 m, with a 0.1 m resolution and a 1.8 m angular step. Dead-reckoning can be computed using the velocity readings coming from the DVL and the heading data obtained from the AHRS sensor, both asynchronously. Standard deviation for the MSIS sensor was set at ± 0.1 m in range and ± 1.8 deg in angular measurements.

For evaluating the algorithm, a small buoy equipped with a DGPS sensor was attached on top of the AUV, with a vertical separation of half a meter. The AUV-buoy system was rigid and allowed the collection of ground truth data simultaneously with the acoustic data (Fig. 7.2b).

7.3.2 The underwater tunnel dataset

The algorithm was also tested with a second dataset obtained in the natural underwater tunnel that passes under "Illa Ferriol" Island, located in the L' Escala area of Costa Brava, Spain (Fig. 7.3a).

The entrance of the tunnel starts at around 10 m depth and reaches almost 24 m at the end of a corridor 28 m long. Although vertical information is sufficient for an MSIS sensor, the tunnel is still a challenging environment because its walls are highly unstructured and its depth is variable (Fig. 7.3b). The survey mission was carried out using the *Sparus* AUV guided along a 100 m path. As in the marina dataset, the MSIS was configured to scan



a) b)

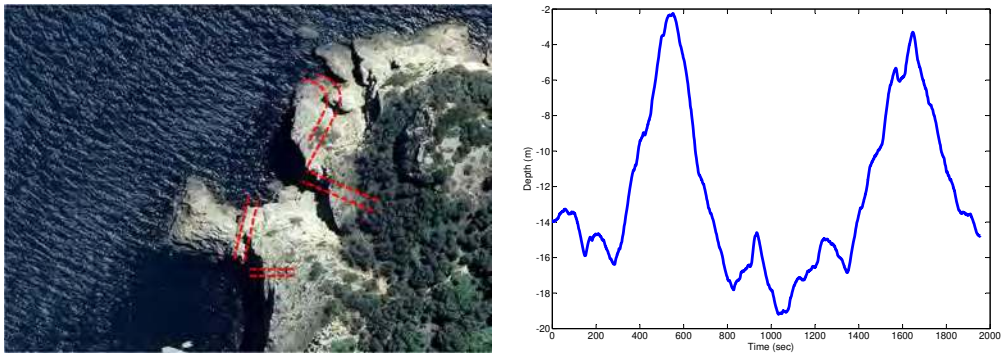
Figure 7.3: Gathering the dataset: a) The approximate position of the underwater tunnel is identified by the dotted red lines. b) The depth profile of the AUV trajectory along the tunnel.

the whole 360 deg sector at 50 m range, with a 0.1 m resolution, 1.8 deg angular step and standard deviations of 0.1 m in range and 1.8 deg in angular measurements. Again, dead-reckoning can be computed using the velocity reading from the DVL and the heading data obtained from the AHRS, both asynchronously. Unfortunately, for this dataset ground truthing is unavailable because the technique of a towed buoy with a GPS unit above the AUV used in the marina dataset cannot be implemented inside a cave.

7.3.3 The underwater caverns dataset

The final test for our algorithm was performed with a dataset obtained in the underwater cavern complex "Coves de Cala Viuda", also located in the L' Escala area of Costa Brava, Spain (Fig. 7.4a). The complex consists of three short single-branch caves and several tunnels of different sizes. This dataset explores two of the caves and tunnels, closing a loop of around 500 m long. The bottom of the caves entrances starts at around 20 m depth, arrives up to surface inside air chambers at the end of the cave, following corridors with diameters varying from 1 to 15 m (Fig. 7.4b).

The dataset was also collected with the *Sparus* AUV guided by a diver all



a) b)

Figure 7.4: Gathering the dataset: a) The approximate position of the underwater caverns are identified by the dotted red lines. b) The depth profile of the AUV trajectory in this dataset.

a) b)

Figure 7.5: a) *Sparus* AUV with extended payload. b) Close-up of the sensors.

the way. This time the MSIS was configured to scan the whole 360 deg sector at 20 m range, with a 0.05 m resolution at 1.8 deg angular step which also define the sensor's standard deviations. For this dataset the payload of the *Sparus* AUV was extended with additional sensors (Fig. 7.5a). First, a second AHRS (ADIS16480, Analog Devices) was added inside the pressure housing with better performance than the original one (MTi, Xsens). For collecting imagery from the seafloor, a 3D video system (Hero2, GoPro) together with two HID lights were mounted so they are downward-looking in front of the main camera. The data recording was uncoupled from the vehicle system and time synchronization was achieved by comparing specific frames with the main camera. Finally, in order to capture the 3D shape of the caves a

a) b)

Figure 7.6: Ground truth points for the cavern dataset. In a) *Sparus* AUV is guided over the cone. b) Measuring the distances between cones.

mechanically scanned profiling sonar was mounted in front and parallel to the forward axis of the AUV (Fig. 7.5b). The profiler emits a conical shape beam with 1 deg angle at 1.1 MHz and was configured to scan the whole 360 deg sector at 10 m range, with a 0.2 m resolution and 1.8 deg angular step. In that configuration, a full sector scan of the profiler takes around 4 sec.

For algorithm validation, the dataset provides six ground truth points in the form of traffic cones (Fig. 7.6a). The cones were in characteristic places along the AUV trajectory over which the vehicle passed twice, including at the entrances of the caves, and at the beginning and end of the survey. The relative position of the cones can be found by extracting the timestamps from the video frames where the cones appear more close to the center of the image and compare them with the timestamps of the trajectory. Then the distance of the cone from the reference frame of the vehicle can be found by using the existing calibration of the camera. An additional ground truthing step was to manually measure and record with tape meter the distances between the cones. However, due to low visibility, long distances and the boulders between the cones, the tape was not always following straight line and these measurements are the upper bound of unknown variance (Fig. 7.6b). Although not very accurate, the tape measurements provide a good sense of scale.

7.4 Results

7.4.1 Results from the marina dataset

Fig. 7.7 shows the trajectory and the map estimated with the proposed SLAM algorithm. In the insert, results are projected on an orthophoto map of the real environment. As expected, the dead-reckoning estimated trajectory suffers from a significant drift which is drastically limited by the SLAM algorithm. The inherent drift of the dead-reckoning is significantly increased in our experiment by the combination of the low accuracy sensors (AHRS, DVL) that the vehicle was equipped with, and the strong presence of ferrous materials (cement walls) in the marina environment. These kinds of perturbations and biases are very difficult to model in advance and are generally treated during post-processing.

Nevertheless, our algorithm was able to correct the bias without any post-processing. The consistency of the algorithm can also be appreciated also visually in the orthophoto map where the scan points and the trajectory are superimposed. The scan points fit the marina walls, except in those areas where the drift is higher (to the top right of the big tank, and towards the left at the end of the trajectory).

The whole dataset was acquired in 53 min with the vehicle traveling at a 0.2 m/sec average speed. The off-line execution of the proposed algorithm, implemented in MATLAB, takes around 14 min with a simple Pentium M @ 2.00 GHz laptop, which is 3.8 times shorter than the duration of the real experiment. It is reasonable to assume that an optimized implementation should be able to operate efficiently on-board the AUV.

The absolute scan displacement error for all scan poses can be seen in Fig. 7.8a. It is computed as the difference between the displacement of a scan estimated through the SLAM algorithm and the corresponding displacement measured with the DGPS, assuming zero DGPS error. Fig. 7.8b and 7.8c shows the histograms of the errors for the X and Y vectors and the solid line

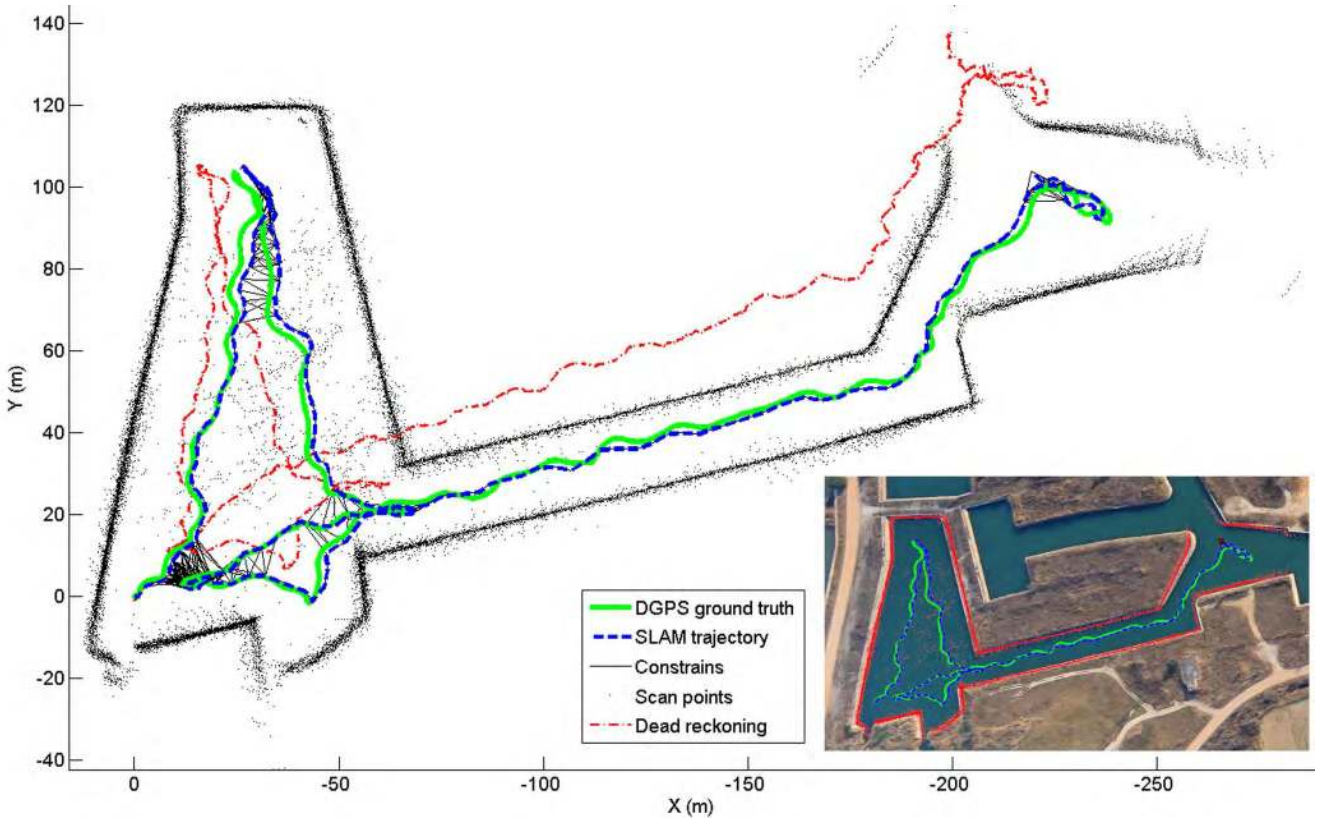


Figure 7.7: SLAM trajectory and map. The red dash-dot line represents the dead-reckoning, the green solid line the DGPS trajectory used as a ground truth and the blue dash line the trajectory estimated with the SLAM algorithm. The insert shows the trajectory and map projected on an orthophoto image.

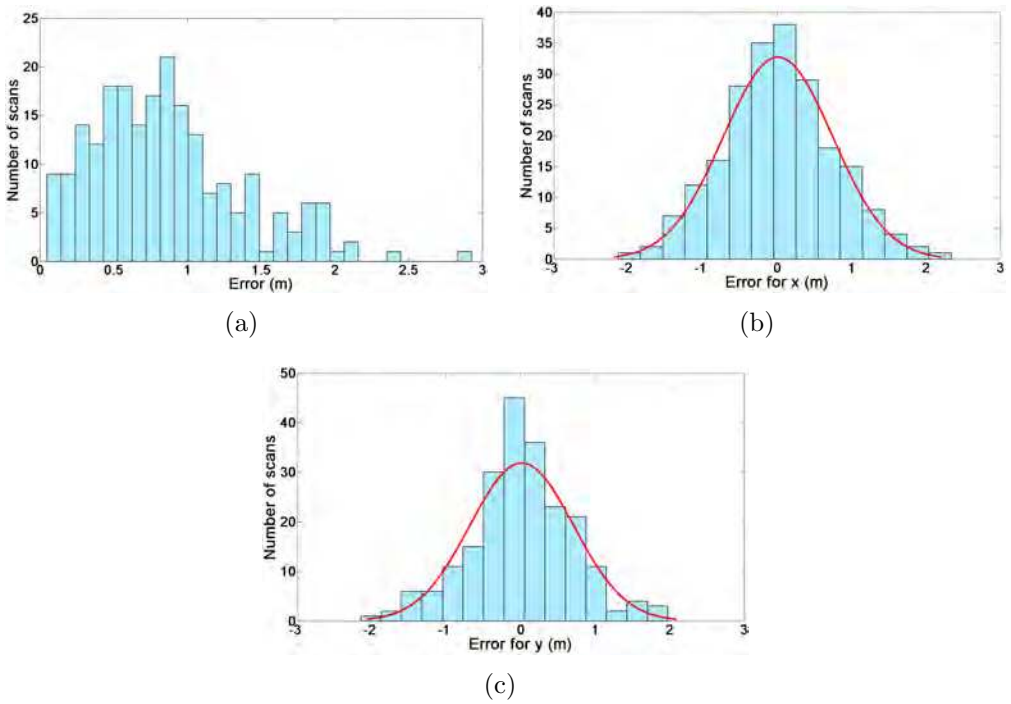


Figure 7.8: Histograms of the error for all the scan displacements estimated through the SLAM and the corresponding displacement measured with the DGPS. (a) Absolute scan displacement error. (b) Error in X axis. (c) Error in Y axis. The solid line is the Gaussian fit to the histogram.

is the Gaussian fit to the histogram.

Fig. 7.9 provides the absolute trajectory error between the results of dead-reckoning, pure scan matching (Hernández et al., 2009), feature-based SLAM (Ribas et al., 2008) and pose-based scan matching SLAM which is the current work. The absolute trajectory error is computed as the difference between the pose estimated through the SLAM algorithm and the corresponding pose measured estimated with the DGPS, assuming zero DGPS error. Table 7.2 provides a summary of the mean, the standard deviation, the minimum and maximum of the absolute trajectory error for the methods we tested. As expected, both SLAM algorithms outperform the pure scan matching implementation. The proposed pose-based scan matching SLAM results are similar to those of the feature-based SLAM. However, the main advantage of our algorithm is that it does not rely on features or any structural information,

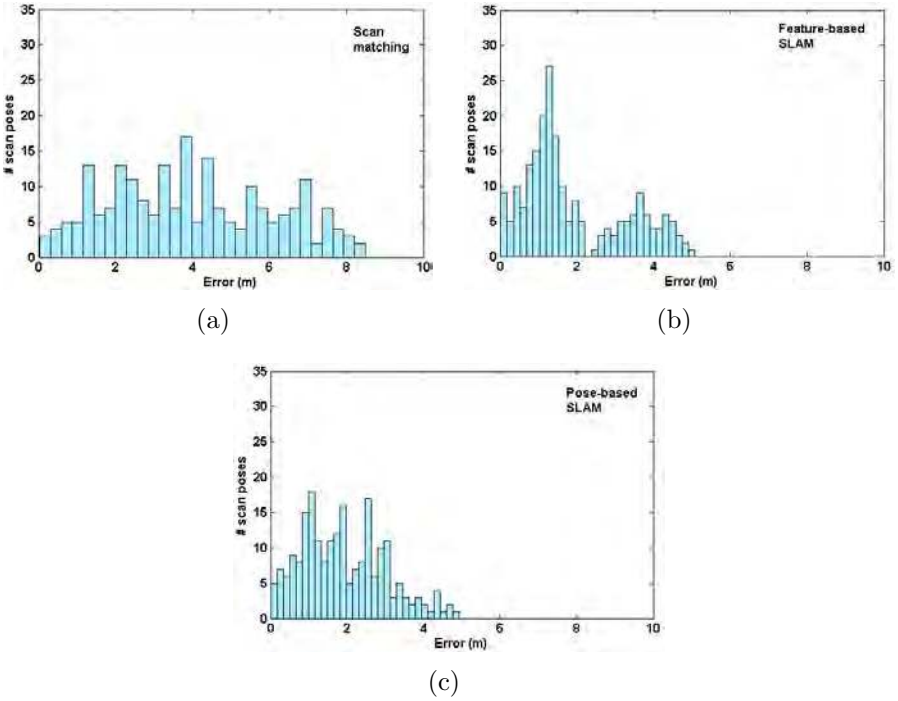


Figure 7.9: Histograms of the absolute trajectory errors for: (a) pure scan matching, (b) feature-based SLAM and (c) pose-based SLAM.

and, therefore, has the potential to be applied in a natural environment.

Table 7.2: Error analysis between the current and previous algorithms

Method / Error (m)	Mean	Std. Dev.	Min.	Max.
Dead-reckoning	17.46	11.47	0.09	45.42
Scan matching	3.99	2.11	0.03	8.49
Feature-based SLAM	1.91	1.32	0.01	5.08
Pose-based SLAM	1.90	1.09	0.01	4.93

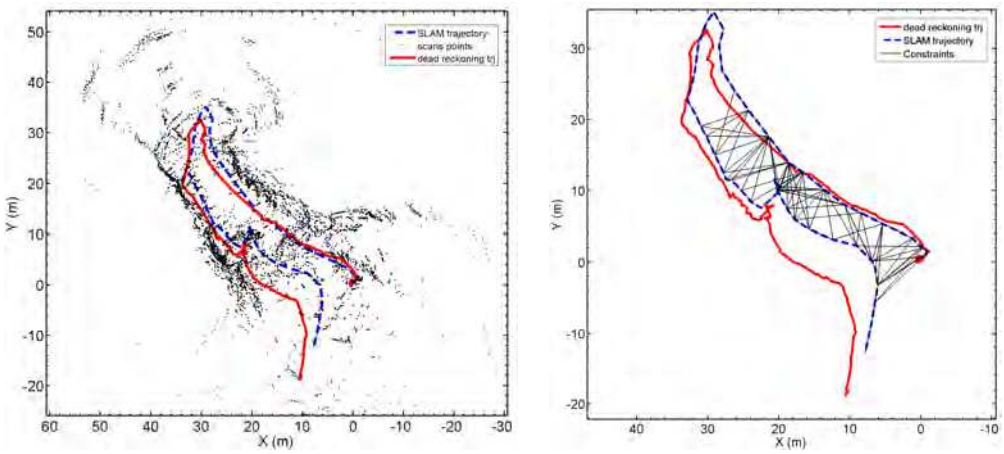
We conclude the results with a note about the ground truth. The nominal accuracy of a DGPS is around 1 m and it degrades at an approximate rate of 0.22 m for each 100 km distance from the broadcast site (Monteiro et al., 2005). In this experiment, the vehicle was receiving differential corrections from a nearby base station (<40 km), therefore, the theoretical DGPS drift is

less than 1.22 m.

7.4.2 Results from the tunnel dataset

This section shows the results of the proposed algorithm when tested in the challenging and highly unstructured environment of a natural underwater tunnel. The test was kept sort in length in order to provide an initial evaluation of the algorithm's performance in such an environment. The vertical beam width of the MSIS is 40 deg, which eliminates the vertical resolution on the sensor (2D scans), but had not introduced any concern of the previous dataset in the marina environment, because the man-made wall was straight and vertical. In this natural dataset, the rough and unstructured walls, the rock debris inside, as well as the vertical extension of the tunnel, produce sonar images that change dramatically from scan to scan. As a consequence, fewer points will be associated between the scans and the registration process will be prone to errors. However, the proposed algorithm was able to cross-register scan poses when the robot was revisiting the same areas on the way back. Fig. 7.10a shows the trajectory and the map estimated with the proposed SLAM algorithm, and Fig. 7.10b shows the constraints between the scan poses. As was expected, there is a drift in the dead-reckoning estimated trajectory which is limited by the proposed SLAM algorithm.

Despite the absence of ground truth data, the relative small area that the cave extends suggests that the trajectory estimate is close to the real trajectory. Visual inspection of the registered scans after the algorithm ends confirms that. An example can be seen in Fig. 7.11. The figure examines scan #10, on the way in, which corresponds to scan #43 on the way out. When the corresponding scans are plotted over the dead-reckoning, the scan points on the way in (black circle) do not coincide with the ones when the AUV revisits the area on the way out (green dots). This phenomenon is limited when the two scans are plotted over the SLAM solution. Another confirmation is obtained coming by scuba diving into the cave. As an example, we measured the entrance of the cave to be around 8 m, which is much smaller than the approximate 15 m that the dead-reckoning suggests. Unfortunately, in such



a) b)

Figure 7.10: SLAM Trajectory and map. The red solid line represents the dead-reckoning and the blue dash line the trajectory estimated with the SLAM algorithm. a) shows the map of the scan points and, b) shows the map of constraints.

an environment, it is very difficult to obtain a ground truth but we partially address this problem in the caverns dataset that follows in the next section.

This dataset was acquired in around 12 min with the vehicle traveling at a 0.15 m/s average speed. The off-line execution of the proposed algorithm, implemented in MATLAB, needs around 2 min with an Intel Core2 Quad @ 3.00 GHz CPU, which as in the structured environment, is promising for real-time implementation.

7.4.3 Results from the cavern dataset

The results from the short experiment in the underwater tunnel were promising but without ground truth, it is very difficult to properly evaluate the behavior of the algorithm. For this reason we tested our algorithm in the longer and more complex environment of an underwater cavern system with ground truth points. As explained previously in this chapter in the description of the dataset, six traffic cones were placed in the cave system along the AUV trajectory where the vehicle passed twice, providing ground truth points

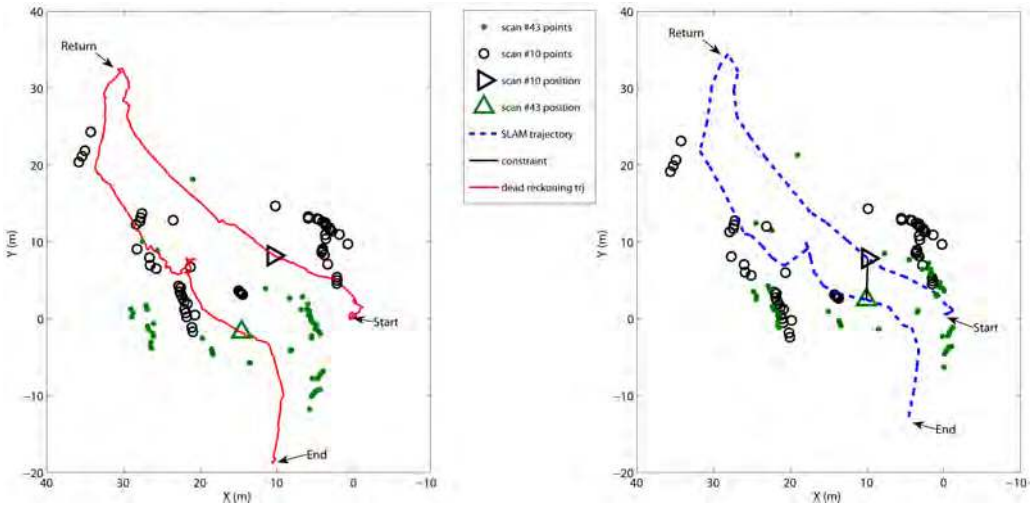


Figure 7.11: Visual inspection of the registered scans #10 and 43. In the left image the scan points are plotted over the dead-reckoning and in the right image over the SLAM solution.

relative to each other.

Because of the higher accuracy of the new AHRS used for this experiment, dead-reckoning estimation was not drifting as much as in the previous datasets. However, there is still enough drift to observe inconsistencies on the resulting map (Fig. 7.12 top), such as the corridor of the top cave and the corridor of the vertical tunnel. Fig. 7.12 (bottom) shows the trajectory and the map estimated with the proposed SLAM algorithm, where these inconsistencies have been greatly reduced. Our algorithm was able to bound the drift by cross-registering most of the scans from areas that the vehicle has visited before. Fig. 7.13 top, shows the constraints from the cross-registration between the scan poses.

The ground truth cones provide two metrics for evaluating the performance of the algorithm relative to the beginning of the experiment.

The first metric, is the difference in distance between the appearances of the same cone when the vehicle passes over it twice. Table 7.3 summarizes these errors for the dead-reckoning (DR) and the SLAM trajectory. Both are visualized in Figure 7.13 (bottom) and with additional detail in Figure 7.14.

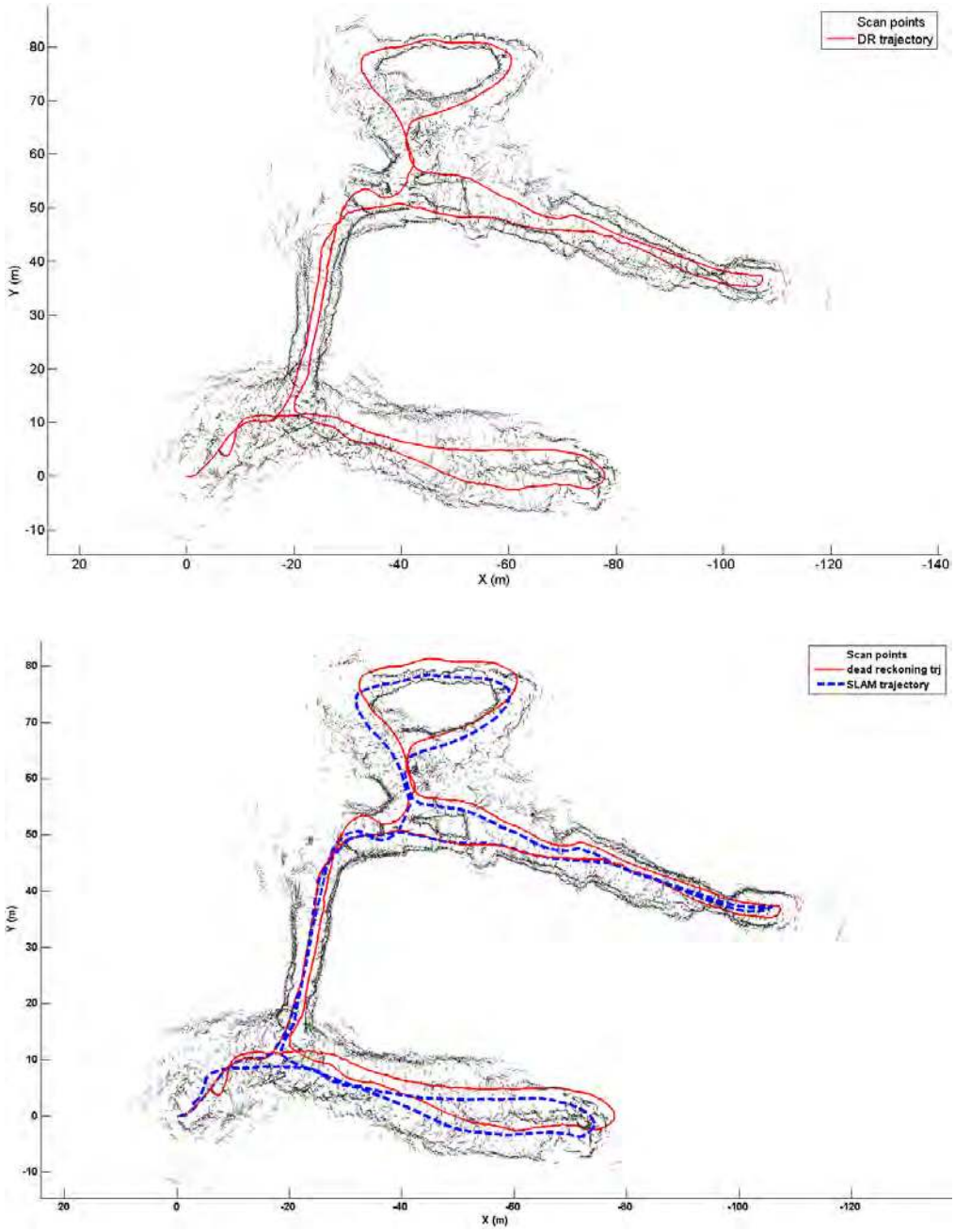


Figure 7.12: Cavern complex trajectories and maps in 2D. Top: map based on dead-reckoning estimation. Bottom: map based on SLAM result.

Table 7.3: Cavern experiment cones errors.

Cones	DR error (m)	SLAM error (m)
1	6.60	2.20
2	3.84	1.68
3	2.81	0.28
4	3.53	0.49
5	2.44	1.40
6	4.37	2.43

The maximum error can be observed at cones 1 and 6. Cone 1 is in an open area where there is not enough vertical information for the scan matching algorithm to identify and close the loop. At cone 6, the algorithm has cross-registered nearby scans; however, the distance from the walls to the vehicle increases the uncertainty of the scans, which results in larger error. For cones located in well confined areas, the algorithm was able to significantly reduce the error. In any case, compared to dead-reckoning, the error reduction varies from half to an order of magnitude.

The second metric is the difference between the distances measured with meter tape by divers from cone to cone, with the ones measured from the estimated trajectories. As explained before, these measurements are the upper bound of an unknown variance due to the fact that it was not always possible the tape to follow a straight line. Although they are not accurate measurements, they provide a good sense of scale. Table 7.4 summarizes all the measured and estimated distances. All the estimated distances are smaller than the measured ones, which demostrate that the algorithm does not suffer from a significant scaling problem. Unfortunately, due to the aforementioned problems, there is no accurate way of identifying which estimated distance is best. However, we can observe that the consistency of measurements between the same pairs has greatly improved over the dead-reckoning estimation.

Additional results validating the algorithm are shown in Figure 7.15, which reproduces two acoustic images generated by placing the sonar measurements

Table 7.4: Cavern experiment error analysis. Cone pairs distance (*see text about ground truth accuracy).

Cone pairs	Ground truth* (m)	DR (m)	SLAM (m)
1 - 2	19	17.07	17.31
2 - 1	19	14.35	16.91
2 - 3	32	31.53	31.00
3 - 2	32	31.38	30.16
3 - 4	16	12.52	12.83
4 - 3	16	13.23	12.87
4 - 5	n.a.	n.a.	n.a.
5 - 4	n.a.	n.a.	n.a.
1 - 6	30	28.36	25.95
6 - 1	30	26.42	26.86

from the complete dataset based on the dead-reckoning and on the SLAM estimated trajectories. An averaged representation of all the overlapping scans has been used; therefore, one can expect the diffuse appearance shown on the dead-reckoning image as a result of the dispersion induced by the erroneous trajectory. On the other hand, using the SLAM trajectory provides a more accurate placement of the measurements which results in a sharper image.

This dataset was acquired in around 32 min and the off-line execution of the proposed algorithm, implemented in MATLAB, needs around 19 min with an Intel Core2 Quad @ 3.00 GHz CPU which, as in the previous experiments, is shows promise for real time implementation.

As explained previously in this chapter in the description of the dataset, the vehicle was also equipped with a profiling sonar scanning 360 deg perpendicular to the traveling direction. We used these scans to build the 3D map of the cavern system as a side-product of our algorithm and as an application example. First, the collected scans were segmented and the range points were extracted as explained in Section 3.2. By projecting the scan points over

the SLAM trajectory (taking into account the depth), a cloud of points was obtained as seen in Figure 7.16 (top).

This cloud of points is the input of the splats distance normalized cut (SDNC) algorithm (Campos et al., 2013) that is used to build the 3D surface model of the cavern system. For each input point, a local quadratic surface was constructed using its k-Nearest Neighbors inside a RANSAC procedure (Fig. 7.16 bottom). These local surfaces were merged in an unsigned distance function evaluated on a tetrahedral grid adapting to the density of the input points. Then, the distance function was signed by using a normalized cut algorithm to segment the volume of the object into its inside and outside. Finally, the surface was extracted at the interface of the two volumes using the restricted Delaunay triangulation surface mesher (Boissonnat and Oudot, 2005).

The final three dimensional reconstruction is shown in the interactive Figure 7.17. If viewed with Acrobat reader 9 and above, the figure can be rotated by holding the left mouse button, zoom with the right button or scroll wheel and panning by holding both buttons. More options for image manipulation appear by right clicking on it.

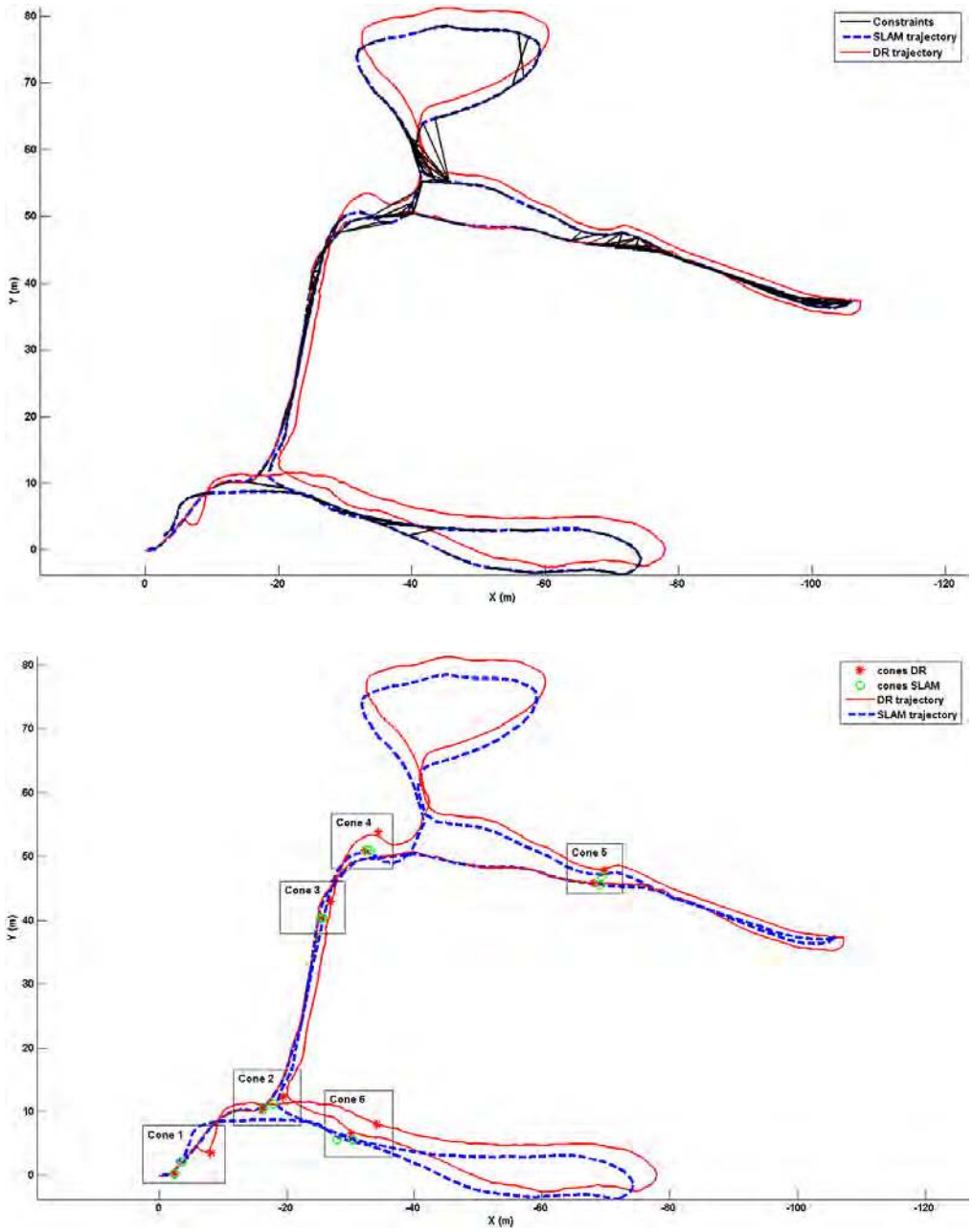


Figure 7.13: Cavern complex map of constraints and ground truth points. Top: map of constraints. Bottom: ground truth points as estimated from dead-reckoning (red crosses) and from SLAM (green circles).

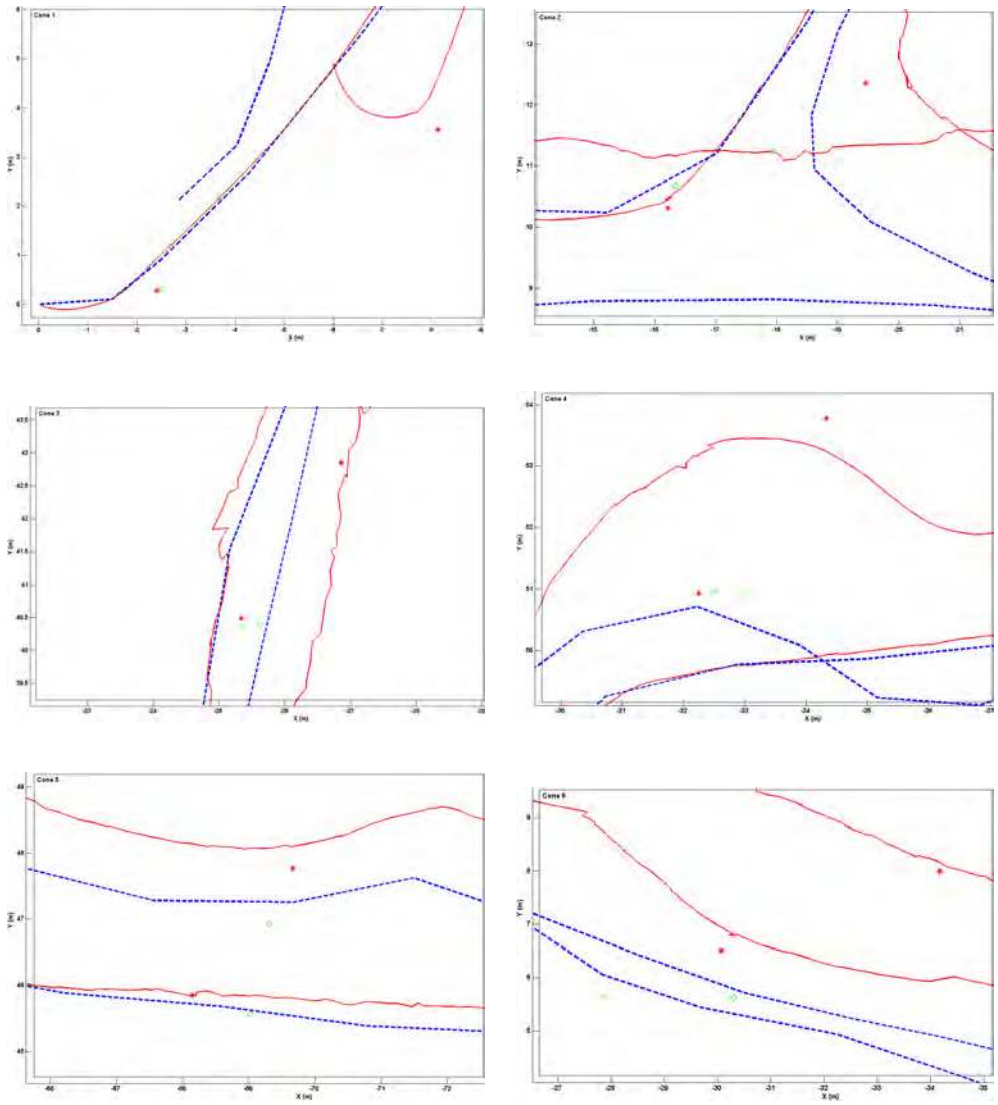


Figure 7.14: Zoom in at the cone areas of Figure 7.13 bottom. Ground truth points as estimated from dead-reckoning (red crosses) and from SLAM (green circles).

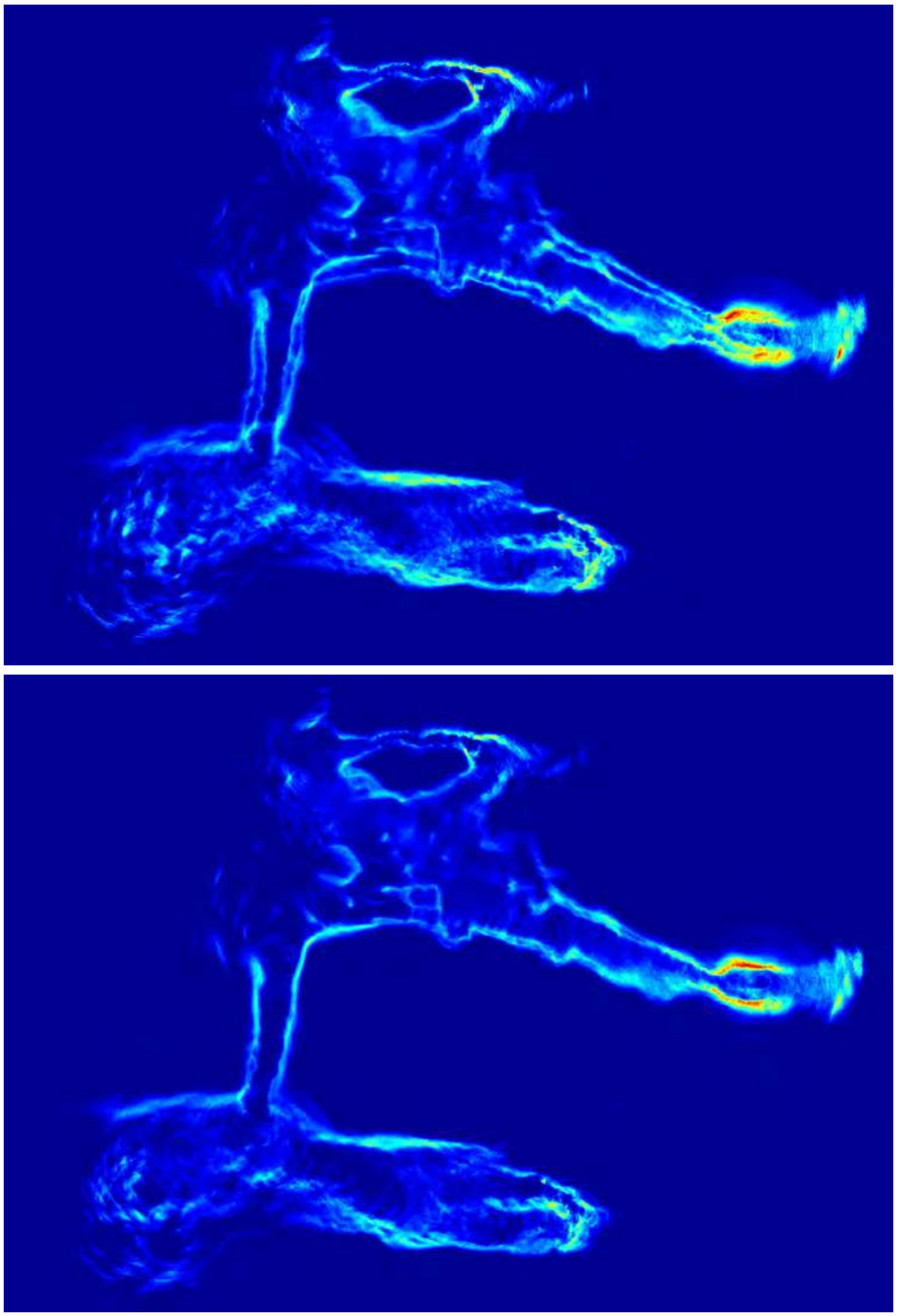


Figure 7.15: Acoustic maps obtained after an averaged composition of the sonar readings. Top: projected over the dead-reckoning trajectory. Bottom: projected over the SLAM estimated trajectory.

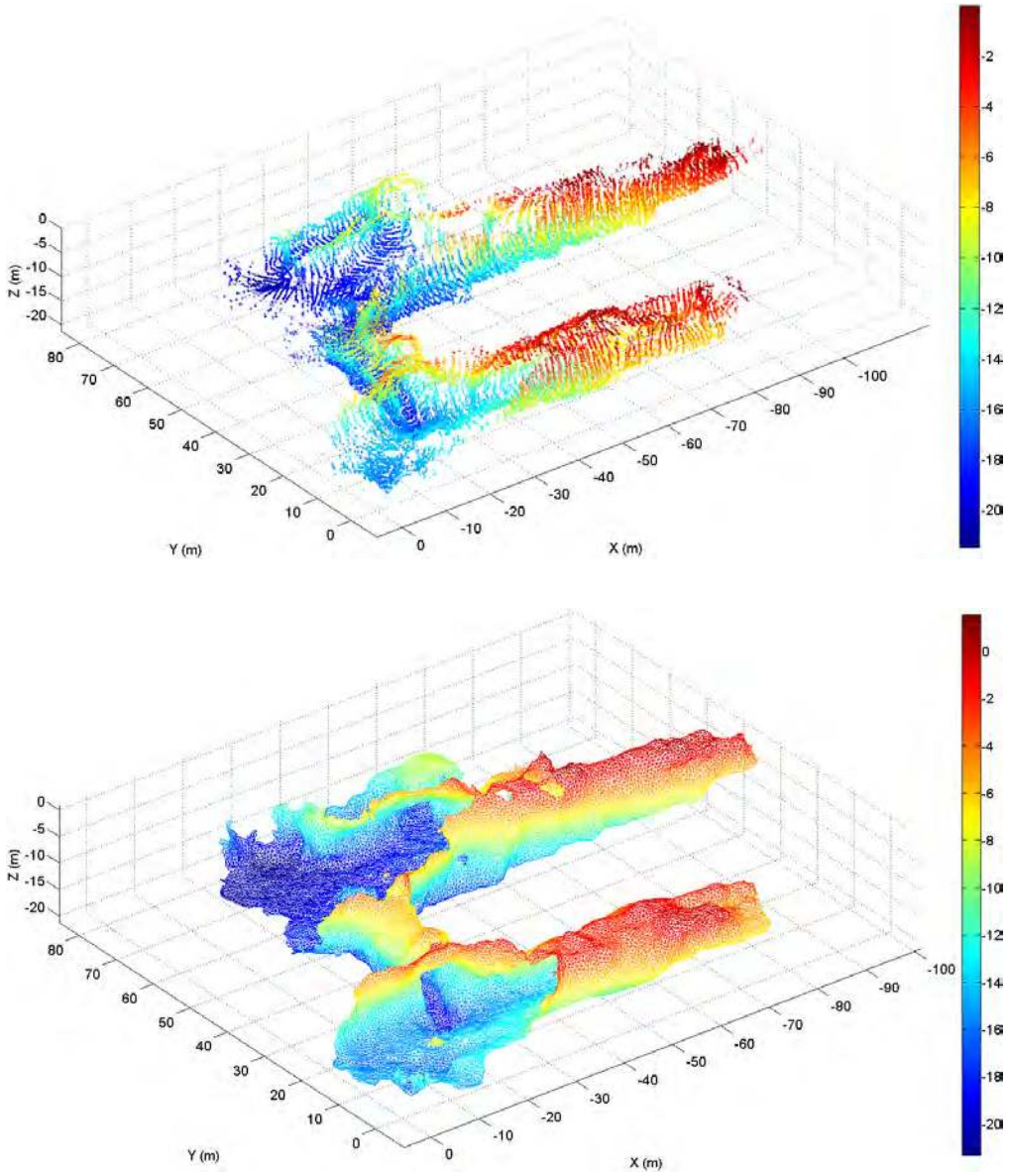


Figure 7.16: Top: Cloud of points of the cavern dataset as extracted from the profiling sonar and projected over the SLAM trajectory. Bottom: Meshed map using the SDNC algorithm. Color map represents depth in meters.

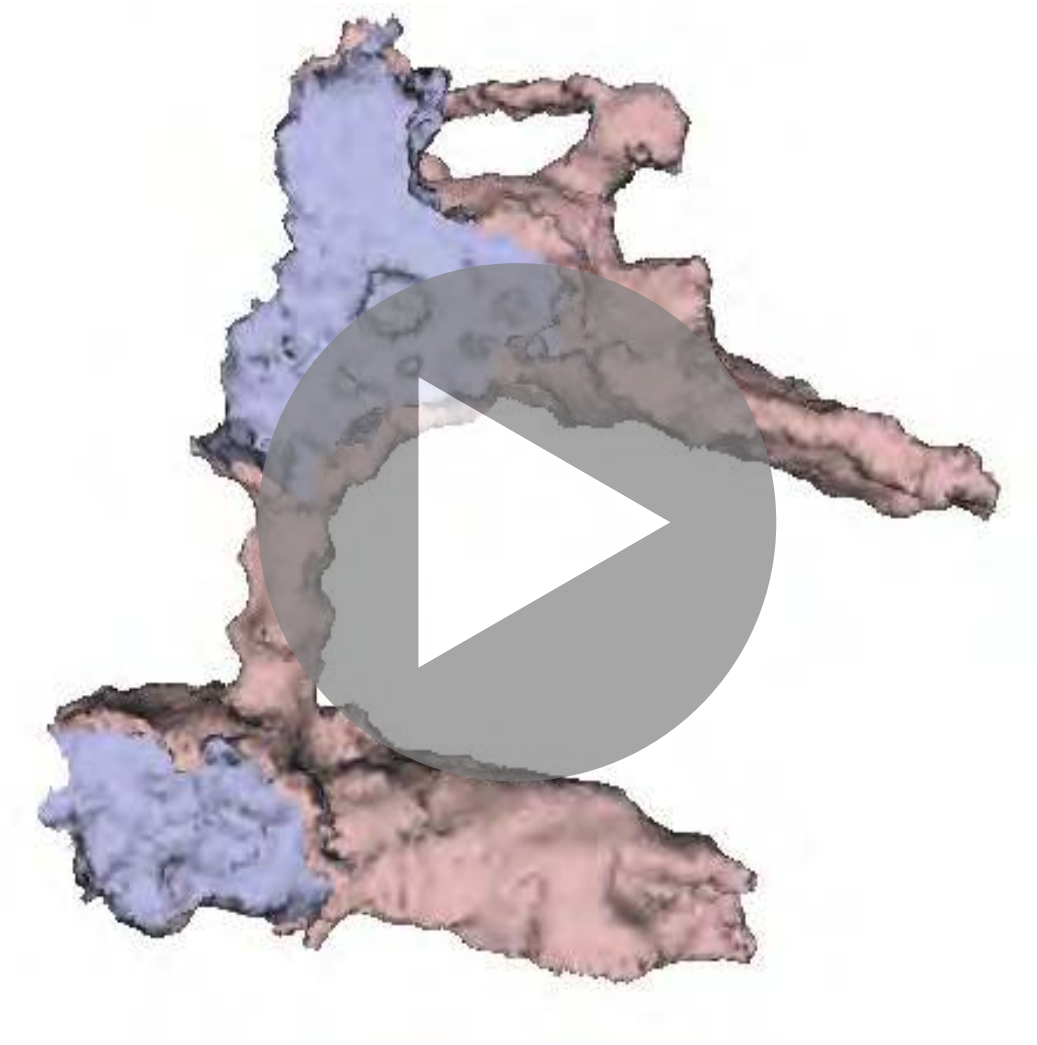


Figure 7.17: Cavern complex 3D surface map. The surface was extracted using the restricted Delaunay triangulation surface mesher *(best viewed with Adobe Acrobat reader 9 and above, left click to rotate the image, right click for additional controls).*

Chapter 8

Discussion and future work

Despite the recent advances in autonomous underwater vehicle navigation techniques, in order to bound the position error, either a set of transponders has to be deployed in the nearby area or the vehicle has to regularly surface to obtain a GPS-fix. However, in confined environments such as underwater caves, surfacing or pre-deployment of localization equipment is not always feasible. This thesis presents a localization and mapping algorithm for an AUV operating in confined environments using a mechanically scanned imaging sonar as the main perception sensor.

8.1 Contributions

The theoretical foundation of this thesis is an augmented state EKF updated from probabilistic scan matching results in a pose-based simultaneous localization and mapping framework. Chapter 2 presents a survey of current conventional underwater localization techniques and the state-of-the-art on SLAM algorithms with a focus on the underwater domain.

The *ScanGrabbing* algorithm receives raw sonar beams acquired by an MSIS while the vehicle is navigating. After an adequate number of beams has been received that cover a full sector, the algorithm forms a scan corrected from the motion-induced distortions and extracts points from the sonar image. Assuming Gaussian noise, the algorithm is able to estimate the uncertainty

of the sonar measurements with respect to a frame located at the center of the scan (Chapter 3).

A survey of the major scan matching algorithms and contributions are examined and categorized in Chapter 4, providing an explanation for the very large number of existing algorithms. The reason for that can be traced to the implementation nature of scan matching. In the same environment, different types of sensors will produce different scans. Based on the specific implementation of this thesis, a comparison proved pIC to be the best among the algorithms.

Calculating the error covariance of a measurement is essential when it has to be combined with other measurements in a probabilistic framework like SLAM. Although ICP-style algorithms are very popular, lead to very good estimates of the relative displacements, and have been improved over the last decade, apparently very little research has attempted to address the uncertainty of their estimates. The first major contribution of this thesis is a closed-form formulation for estimating the uncertainty of the scan matching result (Chapter 5).

For estimating the global trajectory of the vehicle, an ASEKF stores the pose of the vehicle where each full scan was completed. Using the scan-matching algorithm, each new full scan is cross registered with all the previous scans that are in a certain range. This technique has a twofold effect: first, this results in a better estimate of the vehicle's displacement, which is then used to update the ASEKF; and second, loop-closing events are updated automatically and simultaneously (Chapter 6). The algorithm described herein does not take into account any structural information or extract any features. This is the second major contribution of this research effort.

We tested the presented algorithm with three real-world datasets: one obtained in an abandoned marina during an engineering test mission, and two additional ones in the natural environment of underwater cavern systems (Chapter 7). In the marina dataset, the results show the quality of the algorithm by comparing it to the ground truth from a GPS receiver and to other previously published algorithms. For the cavern datasets, the results are compared against fixed ground truth points that the vehicle visits twice along

the trajectory it travels. The cavern datasets are the third major contribution of this research effort, and will be soon available to the scientific community.

8.2 Future work

As a navigation error can lead to the loss of a multi-million euro vehicle, research in navigation algorithms should focus on robustness. The presented work towards in that direction by bounding the navigation error, but it also has allowed us to identify new lines of research.

Scan Correction The presented algorithm is composed of two main filters: one EKF at the local level of a scan estimating beam positions, and an ASEKF for estimating the position of the scans. In that sense, our algorithm has similarities to Hierarchical SLAM (Estrada et al., 2005) where the local submaps (scans in our case) are built to be independent, thus uncorrelated, and the higher level filter keeps the relative positions of the submaps without back-propagating corrections (updates) to them, as the submaps are relatively small. The same reasoning is applicable in this thesis, since the vehicle's motion during the construction of a single scan is very small and so any probable corrections from the higher level would have small effect on the scan precision. As the results show, the presented algorithm performs well with a slow-moving vehicle, which is a reasonable assumption for an open-frame AUV or one operating in confined environments.

Nevertheless, there can be cases where a torpedo-shaped AUV travels with faster speeds and can face the situation where the dead-reckoning estimation drifts significantly while a scan is formed. Future research should consider a way to back-propagate the correction from the SLAM to the scan forming. An adaptation of CI-Graph where the local maps (scans) are conditionally independent and information is shared between the global map and the scans could be one way.

However, we believe that over the next few years, advances in forward looking sonar hardware will provide full 360 deg scan sector instantaneously with high refresh rate, eliminating the scan correction problem.

Scan matching covariance The scan matching covariance estimation is an important element for smooth integration of any probabilistic-based estimator. The closed-form solution that was presented in Chapter 5, is based on Haralick's method (Equation 5.8), where $\mathbf{P}_z$ is a matrix consisting of the uncertainties of the scan points. Theoretically, $\mathbf{P}_z$ can be a full matrix describing the correlations between the scan points, however, practically estimating these correlations is not always trivial. Although in the presented formulation, the $\mathbf{P}_z$ is built as a block diagonal, the results are satisfactory. Further mathematical research will help to include these correlations in a consistent way, which potentially would improve the scan matching covariance estimation.

Simultaneous localization and mapping (SLAM) The proposed pose-based SLAM algorithm uses an ASEKF stochastic map implementation for estimating scan poses. The well-known quadratic expansion problem of the map covariance matrix is partially relaxed in the current application because of the low rate of pose addition (about one per 14 sec) and the short duration of the missions (1-2 hours). Future research will investigate the applicability of other more efficient SLAM frameworks or map representation. To this end, we have already started researching novel map adaptations using octrees (Zandara et al., 2013).

Adding dimensions The mechanically scanned imaging sonar, the main perception sensor for this thesis, provides two-dimensional information which is adequate for areas with walls with low vertical variations, such as in a marina. Our method also performed well in the complex environment of a cave system, however there is still room for improvement as the lack of information on the Z axis produces scans that does not represent the environment accurately. The result is a high rate of wrong or lost registrations which have direct impact to the final trajectory estimation.

Having possibly exploited the maximum capacity of the MSIS sensor, we believe that a native 3D sonar sensor will help build scans that will represent the environment much more accurately. That will avoid most of the false

positive registrations and will register more scans. At the moment only a couple of 3D sonar sensors exist but their size, energy consumption, low resolution and refresh rate prohibit their use with AUVs. Nevertheless, the evolution of sonar sensors suggests that we should expect great improvements over the next few years.

Future work will also research the mix environment cases, where there are also areas without walls. To that end, we currently research methods based on bathymetric (2.5D) sonar that create patches of the seabed from acoustic profiles and cross-registers them as the vehicle moves (Palomer et al., 2013).

Appendix A

The Kalman filter

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

A.1 The linear Kalman Filter

A.1.1 Linear system models

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

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

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

Gaussian probability distribution of covariance $\mathbf{Q}$:

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

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

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

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

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

A.1.2 The Discrete Kalman filter equations

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

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

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

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

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

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

where

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

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

A.2 The Extended Kalman Filter

A.2.1 Non-linear system models

The extended Kalman filter is a version of the Kalman filter that can deal with systems governed by non-linear stochastic difference equations. In this

situation, a non-linear process model is defined as:

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

while a non-linear measurement model is represented as:

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

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

A.2.2 The Discrete Extended Kalman Filter equations

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

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

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

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

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

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

where

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

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

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

Appendix B

Transformations in 2D

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

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

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

B.1 Inversion

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

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

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

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

The Jacobian of the inversion operation is:

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

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

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

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

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

With associated covariance calculated as:

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

B.2 Compounding

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

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

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

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

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

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

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

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

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

with associated covariance approximated as:

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

B.3 Compounding point features

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

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

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

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

The Jacobians of this transformation are:

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

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

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

and its associated covariance as:

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

Bibliography

am Ende, B. (2001, mar/apr). 3D mapping of underwater caves. *Computer Graphics and Applications, IEEE 21*(2), 14 –20.

Aulinas, J., C. Lee, J. Salvi, and Y. Petillot (2010). Submapping slam based on acoustic data from a 6-dof auv. In *8th IFAC Conference on Control Applications in Marine Systems (IFAC/CAMS), Rostock-Warnemnde (Germany)*.

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

Balsamo, A., G. Mana, and F. Pennecchi (2006). The expression of uncertainty in non-linear parameter estimation. *Metrologia 43*(5), 396–402.

Bar-Shalom, Y. and T. E. Fortmann (1988). *Tracking and Data Association*. Boston, MA, USA: Academic Press.

Barkby, S., S. B. Williams, O. Pizarro, and M. V. Jakuba (2011). A featureless approach to efficient bathymetric SLAM using distributed particle mapping. *Journal of Field Robotics 28*(1), 19–39.

BBC News (2013, June). Nigerian cook tells how he survived for three days trapped under sunken ship. http://www.bbc.co.uk/news/world-africa-22888828. Retrieved Dec. 18, 2013.

Bengtsson, O. and A.-J. Baerveldt (2003, June). Robot localization based on scan-matching - estimating the covariance matrix for the IDC algorithm. *Robotics and Autonomous Systems 44*(1), 29 – 40. Best Papers of the Eurobot '01 Workshop.

Besl, P. and H. McKay (1992). A method for registration of 3-D shapes. *IEEE Transactions on pattern analysis and machine intelligence 14*(2), 239–256.

Biber, P. and W. Straßer (2003). The normal distributions transform: A new approach to laser scan matching. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Volume 3, pp. 2743–2748.

Bingham, B., B. Foley, H. Singh, R. Camilli, K. Dellaporta, R. Eustice, A. Mallios, D. Mindell, C. Roman, and D. Sakellariou (2010, November/December). Robotic tools for deep water archaeology: Surveying an ancient shipwreck with an autonomous underwater vehicle. *Journal of Field Robotics 27*(6), 702–714.

Boissonnat, J.-D. and S. Oudot (2005, September). Provably good sampling and meshing of surfaces. *Graph. Models 67*, 405–451.

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

Bosse, M. and J. Roberts (2007). Histogram matching and global initialization for laser-only SLAM in large unstructured environments. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4820–4826.

Bülow, H. and A. Birk (2011). Spectral registration of noisy sonar data for underwater 3D mapping. *Autonomous Robots 30*, 307–331.

Bülow, H. and A. Birk (2013). Spectral 6DOF Registration of Noisy 3D Range Data with Partial Overlap. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*(4), 954–969.

Bülow, H., A. Birk, and V. Unnithan (2009, May). Online generation of an underwater photo map with improved Fourier Mellin based registration. In *OCEANS 2009 - EUROPE*, pp. 1 –6.

Bülow, H., M. Pfingsthorn, and A. Birk (2010). Using robust spectral registration for scan matching of sonar range data. In *7th Symposium on Intelligent Autonomous Vehicles (IAV)*. IFAC.

Bülow, H., I. Sokolovski, M. Pfingsthorn, and A. Birk (2013, June). Underwater stereo data acquisition and 3D registration with a spectral method. In *Proceedings of the MTS/IEEE Oceans Conference, Bergen*, pp. 1–7.

Burgard, W., D. Fox, D. Hennig, and T. Schmidt (1996). Estimating the absolute position of a mobile robot using position probability grids. In *Proceedings of the national conference on artificial intelligence*, pp. 896–901.

Burguera, A., Y. González, and G. Oliver (2007). Probabilistic sonar scan matching for robust localization. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3154–3160.

Burguera, A., Y. González, and G. Oliver (2008). A Probabilistic Framework for Sonar Scan Matching Localization. *Advanced Robotics 22*(11), 1223–1241.

Burguera, A., Y. González, and G. Oliver (2009). On the use of likelihood fields to perform sonar scan matching localization. *Autonomous Robots 26*(4), 203–222.

Burguera, A., G. Oliver, and Y. González (2010). Scan-based SLAM with trajectory correction in underwater environments. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2546–2551.

Campos, R., R. Garcia, P. Alliez, and M. Yvinec (2013). Splat-based surface reconstruction from defect-laden point sets. *Graphical Models 75*(6), 346 – 361.

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

Carreno, S., P. Wilson, P. Ridao, and Y. Petillot (2010). A survey on Terrain Based Navigation for AUVs. In *Proceedings of the MTS/IEEE Oceans conference, Seattle*.

Carrera, A., M. Carreras, P. Kormushev, N. Palomeras, and S. Nagappa (2013). Towards valve turning with an auv using learning by demonstration. In *Proceedings of the MTS/IEEE Oceans conference, Bergen*, pp. 1–7.

Censi, A. (2007, April). An accurate closed-form estimate of ICP's covariance. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Rome, Italy, pp. 3167–3172.

Censi, A. (2008). An ICP variant using a point-to-line metric. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 19–25.

Censi, A., L. Iocchi, and G. Grisetti (2005). Scan matching in the Hough domain. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2739–2744.

Chen, Q., M. Defrise, and F. Deconinck (1994). Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 16*(12), 1156–1168.

Chetverikov, D., D. Svirko, D. Stepanov, and P. Krsek (2002). The Trimmed Iterative Closest Point algorithm. In *16th International Conference on Pattern Recognition.*, Volume 3, pp. 545–548.

Clark, C., C. Olstad, K. Buhagiar, T. Gambin, A. Special, and P. Trust (2008). Archaeology via underwater robots: Mapping and localization within Maltese cistern systems. In *10th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Hanoi, Vietnam,, pp. 662–667.

Davis, D. (1996, September). *Precision maneuvering and control of the Phoenix Autonomous Underwater Vehicle for entering a recovery tube.* Ph. D. thesis, Naval Postgraduate School, Monterey, California.

Diosi, A. and L. Kleeman (2005, August). Laser scan matching in polar coordinates with application to SLAM. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3317–3322.

Douglas-Westwood (2012, February). World AUV Market Forecast 2012-2016.

Duckett, T. (2003). A genetic algorithm for simultaneous localization and mapping. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Volume 1, pp. 434–439.

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

Eliazar, A. I. and R. Parr (2004). DP-SLAM 2.0. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2, pp. 1314–1320.

Elibol, A. (2011). *Efficient Topology Estimation for Large Scale Optical Mapping.* Ph. D. thesis, University of Girona, Doctoral Programme in Technology.

Elibol, A., N. Gracias, and R. Garcia (2010). Augmented state-extended Kalman filter combined framework for topology estimation in large-area underwater mapping. *Journal of Field Robotics 27*(5), 656–674.

Emig, C. and P. Geistdoerfer (2004). The mediterranean deep-sea fauna: historical evolution, bathymetric variations and geographical changes. *Carnets de Géologie / Notebooks on Geology 01*, 1–10.

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

Eustice, R., H. Singh, and J. Leonard (2005). Exactly sparse delayed-state filters. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2417–2424.

Eustice, R., H. Singh, J. Leonard, M. Walter, and R. Ballard (2005). Visually navigating the RMS Titanic with SLAM information filters. In *Proc. Robotics: Science and Systems, MIT Press, Cambridge, MA*, pp. 57–64.

Evans, J., P. Redmond, C. Plakas, K. Hamilton, and D. Lane (2003). Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation. In *Proceedings of the MTS/IEEE Oceans conference, San Diego*, Volume 4, pp. 2201–2210.

Fairfield, N., G. Kantor, and D. Wettergreen (2007). Real-time SLAM with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics 24*(1), 3–21.

Fairfield, N. and D. Wettergreen (2008). Active localization on the ocean floor with multibeam sonar. In *Proceedings of the MTS/IEEE Oceans conference,Quebec City.*

Farr, M. (1991). *The darkness beckons: The history and development of cave diving.* Cave Books.

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

Fleischer, S. (2000). *Bounded-error vision-based navigation of autonomous underwater vehicles.* Ph. D. thesis, Stanford university.

Fock, A. W. (2013, June). Analysis of recreational closed-circuit rebreather deaths 1998–2010. *Diving and Hyperbaric Medicine 43*(2), 78–85.

Foley, B., D. Sakellariou, K. Dellaporta, B. Bingham, R. Camilli, R. Eustice, D. Evagelistis, V. Ferrini, M. Hansson, K. Katsaros, D. Kourkoumelis,

A. Mallios, P. Micha, D. Mindell, C. Roman, H. Singh, D. Switzer, and T. Theodoulou (2009, Apr-Jun). New methods for underwater archaeology: The 2005 Chios ancient shipwreck survey. *HESPERIA - Journal of the American Archaeological School at Athens 78*(2), 269–305.

Fossen, T. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley and Sons Ltd.

François, P., C. Francis, S. Roland, and M. Stéphane (2013). Comparing ICP variants on real-world data sets. *Autonomous Robots 34*(3), 133–148.

Frese, U. (2006). A discussion of simultaneous localization and mapping. *Autonomous Robots 20*(1), 25–42.

Gage, J. and P. Tyler (1991). *Deep-Sea Biology: A Natural History of Organisms at the Deep-Sea Floor*. Cambridge University Press.

Garcia, R., J. Batlle, X. Cufi, and J. Amat (2001). Positioning an underwater vehicle through image mosaicking. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Volume 3.

Garcia, R., X. Cufi, P. Ridao, and M. Carreras (2006). *Robotics and Automation in the Maritime Industries*, Chapter 9: Constructing photo-mosaics to assist UUV navigation and station-keeping, pp. 195–234. AUTOMAR Thematic Network.

German, C., D. Connelly, R. Prien, D. Yoerger, M. Jakuba, A. Bradley, T. Shank, H. Edmonds, and C. Langmuir (2004, December). New techniques for hydrothermal exploration: In situ chemical sensors on AUVs - Preliminary results from the Lau Basin. *EOS: Trans. Amer. Geophys. Union Fall Meet. Supplement 85*(47), A190.

Gracias, N., P. Ridao, R. Garcia, J. Escartín, M. L'Hour, F. Cibecchini, R. Campos, M. Carreras, D. Ribas, N. Palomeras, L. Magi, A. Palomer, T. Nicosevici, R. Prados, R. Hegedüs, L. Neumann, F. de Filippo, and A. Mallios (2013, June). Mapping the Moon: Using a lightweight AUV to survey the site of the 17th century ship 'La Lune'. In *Proceedings of the MTS/IEEE Oceans conference,Bergen*.

Grisetti, G., C. Stachniss, and W. Burgard (2009). Nonlinear constraint network optimization for efficient map learning. *IEEE Transactions on Intelligent Transportation Systems 10*(3), 428–439.

Grisetti, G., G. D. Tipaldi, C. Stachniss, W. Burgard, and D. Nardi (2007, January). Fast and accurate SLAM with Rao-Blackwellized particle filters. *Robotics and Autonomous Systems 55*(1), 30–38.

Gutmann, J. and C. Schlegel (1996). AMOS: comparison of scan matching approaches for self-localizationin indoor environments. In *Proceedings of the First Euromicro Workshop on Advanced Mobile Robot*, pp. 61–67.

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

Haralick, R. (1998). Propagating covariance in computer vision. *Performance Characterization in Computer Vision 1*, 95 – 115.

Hernández, E., P. Ridao, D. Ribas, and J. Batlle (2009). MSISpIC: A probabilistic scan matching algorithm using a mechanical scanned imaging sonar. *Journal on Physical Agents (JoPhA) 3*(1), 3–12.

Huang, S., Z. Wang, and G. Dissanayake (2008). Sparse local submap joining filter for building large-scale maps. *IEEE Transactions on Robotics 24*(5), 1121–1130.

Hurtós, N., D.Ribas, X.Cufí, Y. Petillot, and J. Salvi (2013). Fourier-based registration for robust FLS mosaicing in low visibility underwater environments. *Journal of Field Robotics*.

Johannsson, H., M. Kaess, B. Englot, F. Hover, and J. Leonard (2010, October). Imaging sonar-aided navigation for autonomous underwater harbor surveillance. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4396–4403.

Johnson-Roberson, M., O. Pizarro, S. Williams, and I. Mahon (2010). Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys. *Journal of Field Robotics 27*(1), 21–51.

Kaess, M., H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert (2012, February). iSAM2: Incremental smoothing and mapping using the Bayes tree. *The International Journal of Robotics Research 31*(2), 216–235.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering 82*(1), 35–45.

Kinsey, J., R. Eustice, and L. Whitcomb (2006). A survey of underwater vehicle navigation: Recent advances and new challenges. In *Proceedings of the 7th Conference on Maneuvering and Control of Marine Craft (MCMC). IFAC, Lisbon*.

Konolige, K., G. Grisetti, R. Kummerle, W. Burgard, B. Limketkai, and R. Vincent (2010). Efficient sparse pose adjustment for 2D mapping. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 22–29.

Kümmerle, R., G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard (2011). G2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3607–3613.

Kunz, C., C. Murphy, H. Singh, C. Pontbriand, R. Sohn, S. Singh, T. Sato, C. Roman, K. Nakamura, M. Jakuba, et al. (2009). Toward extraplanetary under-ice exploration: Robotic steps in the Arctic. *Journal of Field Robotics 26*(4), 411–429.

Kunz, C. and H. Singh (2013, September). Map building fusing acoustic and visual information using autonomous underwater vehicles. *Journal of Field Robotics 30*(5), 763–783.

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

Lenac, K., E. Mumolo, and M. Nolich (2007). Fast genetic scan matching using corresponding point measurements in mobile robotics. In M. Giacobini (Ed.), *Applications of Evolutionary Computing*, Volume 4448 of *Lecture Notes in Computer Science*, pp. 375–382. Springer Berlin Heidelberg.

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

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

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

Lu, F. and E. Milios (1994). Robot pose estimation in unknown environments by matching 2D range scans. In *Proceedings of IEEE Computer Society*

Conference on Computer Vision and Pattern Recognition (CVPR), pp. 935–938.

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

Lu, F. and E. Milios (1997b, March). Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent & Robotic Systems 18*(3), 249–275.

Magnusson, M., A. Lilienthal, and T. Duckett (2007). Scan registration for autonomous mining vehicles using 3D-NDT. *Journal of Field Robotics 24*(10), 803–827.

Mahalanobis, P. C. (1936). On the generalized distance in statistics. In *Proceedings of the National Institute of Science of India*, Volume II, pp. 49–55.

Mahon, I., O. Pizarro, M. Johnson-Roberson, A. Friedman, S. B. Williams, and J. C. Henderson (2011). Reconstructing Pavlopetri: mapping the world's oldest submerged town using stereo-vision. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2315–2321.

Mahon, I., S. Williams, O. Pizarro, and M. Johnson-Roberson (2008, October). Efficient view-based SLAM using visual loop closures. *IEEE Transactions on Robotics 24*(5), 1002–1014.

Maki, T., H. Kondo, T. Ura, and T. Sakamaki (2006, September). Photo mosaicing of Tagiri shallow vent area by the AUV "Tri-Dog 1" using a SLAM based navigation scheme. In *Proceedings of the MTS/IEEE Oceans conference, Boston.*

Marani, G., S. Choi, and J. Yuh (2009). Underwater Autonomous Manipulation for Intervention Missions AUVs. *Ocean Engineering. Special Issue: AUV 36*(1), 15–23.

Martínez, J., J. González, J. Morales, A. Mandow, and A. García-Cerezo (2006). Mobile robot motion estimation by 2D scan matching with genetic and iterative closest point algorithms. *Journal of Field Robotics 23*(1), 21–34.

Masuda, T. and N. Yokoya (1994). A robust method for registration and segmentation of multiple range images. In *Proceedings of the Second IEEE CAD-Based Vision Workshop*, pp. 106–113.

Maybeck, P. (1982). *Stochastic models, estimation, and control*, Volume 141-1 of *Mathematics in Science and Engineering*. Academic Press.

McVicker, W., J. Forrester, T. Gambin, J. Lehr, Z. Wood, and C. Clark (2012). Mapping and visualizing ancient water storage systems with an ROV – an approach based on fusing stationary scans within a particle filter. In *Proceedings of IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 538–544.

Medwin, H. and C. S. Clay (1998). *Fundamentals of Acoustical Oceanography*. Boston: Academic Press.

Milne, P. (1983). *Underwater acoustic positioning systems*. Gulf Publishing Co., Houston, TX.

Mínguez, J., F. Lamiraux, and L. Montesano (2005, April). Metric-based scan matching algorithms for mobile robot displacement estimation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3557–3563.

Monteiro, L., T. Moore, and C. Hill (2005). What is the accuracy of DGPS? *The Journal of Navigation 58*(02), 207–225.

Montemerlo, D., N. Roy, and S. Thrun (2003). Perspectives on standardization in mobile robot programming: the Carnegie Mellon navigation (CARMEN) toolkit. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Volume 3, pp. 2436–2441.

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

Montesano, L., J. Minguez, and L. Montano (2005, August). Probabilistic scan matching for motion estimation in unstructured environments. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3499–3504.

Newman, K. R., M.-H. Cormier, J. K. Weissel, N. W. Driscoll, M. Kastner, H. Singh, R. Camilli, R. Eustice, N. McPhee, J. C. Ahern, G. Robertson, E. Solomon, and K. Tomanka (2004). A detailed near-bottom survey of large gas blowout structures along the US Atlantic shelf break using the autonomous underwater vehicle (AUV) SeaBED. In *EOS: Transactions of*

the American Geophysical Union Fall Meeting Supplement, Volume 85, pp. 1312. Abstract.

Newman, P. M. (1999, March). *On the structure and solution of the simultaneous localisation and map building problem*. Ph. D. thesis, Australian Centre for Field Robotics. The University of Sydney.

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

Newman, P. M., J. J. Leonard, and R. J. Rikoski (2003, October). Towards constant-time SLAM on an autonomous underwater vehicle using synthetic aperture sonar. In *Proceedings of the 11th International Symposium on Robotics Research*, Sienna, Italy, pp. 409–420.

Ni, K. and F. Dellaert (2010). Multi-level submap based SLAM using nested dissection. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2558–2565.

Nieto, J., T. Bailey, and E. Nebot (2007, January). Recursive scan-matching SLAM. *Robotics and Autonomous Systems 55*(1), 39–49.

NOAA (2013). Oceans. http://www.noaa.gov/ocean.html. Retrieved Dec. 5, 2013.

Nygren, I. (2005). *Terrain navigation for underwater vehicles*. Ph. D. thesis, KTH, Signals, Sensors and Systems.

Olson, E., J. Leonard, and S. Teller (2006). Fast iterative alignment of pose graphs with poor initial estimates. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2262–2269.

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

Palomer, A., P. Ridao, D. Ribas, A. Mallios, N. Gracias, and G. Vallicrosa (2013). Bathymetry-based SLAM with Difference of Normals Point-Cloud subsampling and probabilistic ICP registration. In *Proceedings of the MTS/IEEE Oceans conference, Bergen*.

Paskin, M. A. (2003). Thin junction tree filters for simultaneous localization and mapping. In G. Gottlob and T. Walsh (Eds.), *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*, San Francisco, CA, pp. 1157–1164. Morgan Kaufmann Publishers.

Paz, L. M., J. Tardos, and J. Neira (2008). Divide and Conquer: EKF SLAM in O(n). *IEEE Transactions on Robotics 24*(5), 1107–1120.

Pfingsthorn, M., A. Birk, and H. Bülow (2012, May). Uncertainty estimation for a 6-DoF spectral registration method as basis for sonar-based underwater 3D SLAM. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3049–3054.

Pfingsthorn, M., A. Birk, S. Schwertfeger, H. Bülow, and K. Pathak (2010, May). Maximum likelihood mapping with spectral image registration. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4282–4287.

Pfister, S., K. Kriechbaum, S. Roumeliotis, and J. Burdick (2002). Weighted range sensor matching algorithms for mobile robot displacement estimation. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2, pp. 1667–1674.

Piniés, P., L. Paz, D. Gálvez-López, and J. Tardós (2010). CI-Graph simultaneous localization and mapping for three-dimensional reconstruction of large and complex environments using a multicamera system. *Journal of Field Robotics 27*(5), 561–586.

Pizarro, O., A. Friedman, M. Johnson-Roberson, S. Williams, I. Mahon, R. Camilli, L. Camilli, and A. Mallios (2010, September). High-resolution optical and acoustic 3D seafloor reconstructions from robot and diver-based surveys. In *XIX Congress of the Carpathian-Balkan geological association*, pp. 314. Carpathian-Balkan Geological Association. Abstracts volume.

Pizarro, O., H. Singh, and S. Lerner (2001). Towards image-based characterization of acoustic navigation. In *Intelligent Robots and Systems (IROS)*.

Reddy, B. and B. Chatterji (1996). An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Transactions on Image Processing 5*(8), 1266–1271.

Reynolds, J., R. Highsmith, B. Konar, C. Wheat, and D. Doudna (2001). Fisheries and fisheries habitat investigations using undersea technology. In *Proceedings of the MTS/IEEE Oceans conference, Honolulu*, Volume 2, pp. 812–820.

Ribas, D., N. Palomer, P. Ridao, M. Carreras, and E. Hernandez (2007, apr). Ictineu AUV wins the first SAUC-E competition. In *IEEE International Conference on Robotics and Automation (ICRA)*, Roma, Italy, pp. 151–156.

Ribas, D., N. Palomeras, P. Ridao, M. Carreras, and A. Mallios (2012, feb.). Girona 500 auv: From survey to intervention. *IEEE/ASME Transactions on Mechatronics 17*(1), 46 –53.

Ribas, D., P. Ridao, J. Tardós, and J. Neira (2008, July). Underwater SLAM in man made structured environments. *Journal of Field Robotics 25*(11-12), 898–921.

Ridao, P., M. Carreras, D. Ribas, P. J. Sanz, and G. Oliver (2014). Intervention AUVs: The Next Challenge. In *IFAC world Congress. Keynote in the Marine Robotics Workshop.*

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

Rusinkiewicz, S. and M. Levoy (2001). Efficient variants of the ICP algorithm. In *Proceedings of Third International Conference on 3-D Digital Imaging and Modeling*, pp. 145–152.

Salvi, J., Y. Petillot, and E. Batlle (2008). Visual SLAM for 3D large-scale seabed acquisition employing underwater vehicles. In */proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1011–1016.

Schofield, O., J. Kohut, D. Aragon, and L. Creed (2007). Slocum gliders: Robust and ready. *Journal of Field Robotics 6*(24), 473–485.

Singh, H., R. Armstrong, F. Gilbes, R. Eustice, C. Roman, O. Pizarro, and J. Torres (2004). Imaging coral I: Imaging coral habitats with the SeaBED AUV. *Subsurface Sensing Technologies and Applications 5*(1), 25–42.

Singh, H., C. Roman, O. Pizarro, R. Eustice, and A. Can (2007). Towards high-resolution imaging from underwater vehicles. *The International Journal of Robotics Research 26*(1), 55–74.

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

Stone, W., B. Hogan, C. Flesher, S. Gulati, K. Richmond, A. Murarka, G. Kuhlman, M. Sridharan, V. Siegel, R. Price, et al. (2010). Design and deployment of a four-degrees-of-freedom hovering autonomous underwater vehicle for sub-ice exploration and mapping. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment 224*(4), 341–361.

Stutters, L., H. Liu, C. Tiltman, and D. Brown (2008). Navigation technologies for autonomous underwater vehicles. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 38*(4), 581–589.

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

Tena, I. (2001, September). *Enhanced Concurrent Mapping and Localisation Using Forward-looking Sonar*. Ph. D. thesis, Heriot-Watt University.

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

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

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

Thrun, S., W. Burgard, and D. Fox (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.

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

Thrun, S. and M. Montemerlo (2006, May). The graph SLAM algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research 25*(5-6), 403–429.

Trucco, E., A. Fusiello, and V. Roberto (1999). Robust motion and correspondence of noisy 3-D point sets with missing data. *Pattern recognition letters 20*(9), 889–898.

Tuoy, S. T., T. Maekawa, and N. M. Patrikalalis (1993). Interrogation of geophysical maps with uncertainty for AUV micro-navigation. In *Proceedings of the MTS/IEEE Oceans conference, Victoria*, pp. I181–I185.

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

Walter, M., F. Hover, and J. Leonard (2008). SLAM for ship hull inspection using exactly sparse extended information filters. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1463–1470.

Weiß, G. and E. von Puttkamer (1995). A map based on laser scans without geometric interpretation. In *Proceedings of the International Conference on Intelligent Autonomous Systems*, Karlsruhe, Germany, pp. 403–407.

Williams, B., G. Klein, and I. Reid (2007). Real-time SLAM relocalisation. In *International Conference on Computer Vision (ICCV)*.

Williams, S. and I. Mahon (2004). Simultaneous localisation and mapping on the great barrier reef. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2, pp. 1771–1776.

Williams, S. B., G. Dissanayake, and H. F. Durrant-Whyte (2002, May). An efficient approach to the simultaneous localisation and mapping problem. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 1, Washington DC, USA, pp. 406–411.

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

Zandara, S. and A. Nicholson (2009). Square root unscented particle filtering for grid mapping. In A. Nicholson and X. Li (Eds.), *AI 2009: Advances in Artificial Intelligence*, Volume 5866 of *Lecture Notes in Computer Science*, pp. 121–130. Springer Berlin Heidelberg.

Zandara, S., P. Ridao, D. Ribas, A. Mallios, and A. Palomer (2013). Probabilistic Surface Matching for Bathymetry Based SLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 40–45.

Ze-Su, C., H. Bing-Rong, and L. Hong (2007). An improved polar scan matching using genetic algorithm. *Information Technology Journal 6*(1), 89–95.

Zhang, F., X. Chen, M. Sun, M. Yan, and D. Yang (2004). Simulation study of underwater passive navigation system based on gravity gradient. In *Proceedings of IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Volume 5, pp. 3111–3113.

Zinßer, T., J. Schmidt, and H. Niemann (2003). A refined ICP algorithm for robust 3-D correspondence estimation. In *Proceedings of International Conference on Image Processing (ICIP)*, Volume 2, pp. 695–698.

Declaration:

I hereby declare that his thesis contains no material which has been accepted for the award of any other degree or diploma in any university. To the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference has been made.

Girona, April 2014

Angelos Mallios