Absolute Localization for Autonomous Robotic Inspection in Confined Spaces

Brogaard, Rune Yding

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Absolute Localization for Autonomous Robotic Inspection in Confined Spaces

by

Rune Yding Brogaard

Dissertation submitted March 2023

Technical University of Denmark

Department of Electrical and Photonics Engineering
Automation and Control (AUT)
Technical University of Denmark
Elektrovej, Building 326
DK-2800 Kgs. Lyngby
Denmark
www.electro.dtu.dk/
Tel: (+45) 45 25 38 00
E-mail: electro@electro.dtu.dk

"A society grows great when old men plant trees whose shade they know they shall never sit in."

Greek Proverb

Preface

This dissertation was submitted to the Department of Electrical and Photonics Engineering at the Technical University of Denmark in fulfillment of the requirements for a Ph.D. degree. The work presented in this dissertation was funded by FORCE Technology. The project was carried out in collaboration with the Inspectrone project(*Autonomous and high-level commanded system for remote inspection of marine vessels to support classification and commercial operations*), under IFD contract number 8090-00080B.

The dissertation studies autonomous inspection of confined spaces with a focus on localization and marine vessels. Seven academic articles have been written as part of the Ph.D. project. Five of these articles are peer reviewed and published, two have been submitted and are awaiting peer review. The Ph.D. study began in October 2019 and was submitted in March 2023. The project was supervised by:

- Associate Professor Evangelos Boukas (principal supervisor), Department of Electrical and Photonics Engineering, DTU.
- Chief Operating Officer Lars Vesth (co-supervisor), FORCE Technology
- Professor Ole Ravn (co-supervisor), Department of Electrical and Photonics Engineering, DTU.

The research was carried out at the Department of Digital Innovation at FORCE Technology, in the section of Automation and Control at DTU, and during an external stay at NASA Jet Propulsion Laboratory in California under the supervision of Robotics Technologist Robert Hewitt. The dissertation consists of a summary report that describes the contributions of 7 research articles, and includes the articles themselves denoted as Article A-G in the following chapters.

Abstract

The modern world is built upon a large industrial sector consisting of areas such as manufacturing plants, chemical processing, energy production, and a large marine-based global supply chain. Maintaining and ensuring optimal runtime of these large and complex assets requires frequent inspections and repairs. These inspection missions pose a series of potential health risks to human surveyors, and most critically confined space inspections result in several yearly deaths across the industry. It is clear that a robotic solution has the potential to mitigate some of these fatalities by removing humans from the danger zone. Due to the complex shape of the associated operational spaces, teleoperated robots do not perform well. Therefore, autonomous robotic solutions with the ability to self-localize and self-navigate are required. However, the localization of robots in confined industrial spaces poses a range of challenges. The environment is often feature-few, and poorly lit, and a high level of symmetry tends to be a common factor across most sectors. Due to these challenges, autonomous robot-based inspections with the ability to provide absolute positions of the robot and defects are currently non-existent in the inspection of industrial confined spaces. To perform state estimation on autonomous robots, well-formed uncertainty estimates are required to be provided by all individual estimates. Therefore, accurate uncertainty estimation for 3D-3D registrations, in the context of localization within a given map of the environment, is also an area where further research is needed. Moreover, the ability to provide a good pose and uncertainty estimate of the defects within the critical areas of a vessel, is important to provide a proper evaluation of each vessel, whether evaluated by a human or by machine learning approaches. The first step in filling this gap in research and industry is the investigation of the optimal system to obtain absolute localization both for defect pose estimation and as an input for the navigation of the robots themselves. Therefore, the presented research aims to explore viable solutions for absolute localization in autonomous robotic inspection of confined spaces, using a mixture of compact lightweight visual and ToF cameras, with the goal of fitting the resulting system onto payload-limited platforms in the shape of UAVs or small legged robots. The research presented in this dissertation is focused on the water ballast tanks of marine vessels and begins by tackling the absolute localization within these known environments using unmanned aerial vehicles. As part of the first presented method, a custom feature detector was developed to provide a robotic and defect pose estimate in a high-level map of the ballast tanks. The disadvantages of this approach were later addressed by employing more general deep learning-based feature descriptors in combination with a novel registration algorithm, Teaser++. To handle the computational load and ambiguity of the environment, specific attention was given to the design / execution of the developed algorithms onboard GPUs as well as to the inclusion of state-of-the-art uncertainty estimation. In addition, an exploration-based inspection approach with a focus on defect detection was investigated and proved viable in a real-world simulated water ballast tank. With these findings, it was shown that it is possible to use small-scale aerial robotic systems to autonomously inspect confined spaces.

Resumé

Den moderne verden er bygget på en stor industrisektor, bestående af områder såsom produktion- og kemianlæg, energiproduktion og en stor, global maritim forsyningskæde. Vedligeholdelse og driftsikring af disse store og komplekse aktiver kræver hyppige inspektioner og reparationer. Inspektionsopgaverne udgør en række potentielle sundhedsrisici for inspektørerne, og især inspektioner af lukkede rum resulterer i en række årlige dødsfald. En robotbaseret løsning har potentialet til at reducere nogle af disse dødsfald ved at fjerne de menneskelige aktører fra disse risici. Konstruktionen og udformningen af de lukkede rum resulterer dog ofte i, at fjernstyrede robotter i de fleste tilfælde ikke kan anvendes. Derfor er det nødvendigt med autonome robotløsninger der har evnen til at finde deres egen position i rummet og navigere i det svært tilgængelige miljø. Imidlertid udgør lokaliseringen af robotter i lukkede industrielle rum en række udfordringer. Miljøet er ofte kendetegnet ved ensartede overflader og dårlig belysning, samt et højt niveau af symmetri i mange af de industrielle konstruktioner. På grund af disse udfordringer eksistere der i skrivende stund ikke autonome robotbaserede inspektionsløsninger der har evnen til at positionere robotten og detekterede defekter i industrielle lukkede rum. For at positions-bestemme autonome robotter kræves også en velfunderet vurdering af usikkerheden på den estimerede position. Nøjagtigheden af denne usikkerhed er især vigtig for 3D-3D registreringer fra en sensor dannet punktsky, til et givet kort eller CAD-model af området. Ydermere er evnen til at give en god position og vurdering af defekterne inden for de kritiske områder, i eksempelvis et skib, vigtig for at give en korrekt evaluering skibets tilstand — uanset om det vurderes af et menneske eller ved hjælp af kunstig intelligens. Det første skridt i at udfylde dette hul i forskningen og industrien er undersøgelsen af et system til opnåelse af absolut lokalisering — både til estimering af defektposition og som input til selve robotternes navigationssystem. Den præsenterede forskning i denne afhandling har derfor til formål at udforske rentable løsninger til absolutte lokalisering i autonom robotinspektion af lukkede rum. Løsningen vil anvende en blanding af kompakte visuelle- og ToF-kameraer med det formål at tilpasse det resulterende system til platforme med begrænset nyttelast i form af droner eller firbenede robotter. Den forskning der præsenteres i denne afhandling, er primært fokuseret på inspektion af ballast-tanke i marinefartøjer og begynder med at tackle den absolutte positionering inden for disse kendte miljøer ved hjælp af ubemandede luftfartøjer. Som en del af den først præsenterede løsning, blev en rumspecifik metode udviklet til at give positionsestimater af detekterede defekter ved at anvende en detektion af strukturelle elementer i en ballast tank. Ulemperne ved denne rumspecifikke tilgang blev senere løst ved at anvende mere generelle deep learning-baserede metoder i kombination med en ny registreringsalgoritme, Teaser++. Dette resulterer i bredere anvendelsemuligheder på tværs af industrier. For at håndtere de beregningstunge algoritmer, og ensartetheden i det industrielle miljø, blev der tillagt særlig opmærksomhed på at udviklede algoritmer der kunne køre ombord på en GPU, samt på inklusion af nye metoder til vurdering af nøjagtigheden af positionsestimaterne. Derudover blev en udforsknings-baseret inspektionstilgang, med fokus på defektdetektering, undersøgt og vist sig rentabel i en simuleret vandballasttank i den virkelige verden. Med disse resultater blev det vist, at det er muligt at anvende små droner til autonom inspektion af lukkede rum.

Acknowledgements

This dissertation was made possible thanks to the support from FORCE Technology and the Inspectrone Project at the Technical University of Denmark. I am grateful for the trust and support received from the staff and management at both of these institutions. I would like to first and foremost thank the Inspectrone project manager and my supervisor, Evangelos Boukas, as well as my co-supervisors, Lars Vesth and Ole Ravn. A special thanks to Evangelos Boukas for your impressive commitment and late-night dedication. This has gone beyond anyone's expectations throughout my study, and I am truly thankful for this. We have experienced many interesting and challenging tasks, resulting in both professional and personal growth, which would not have been possible without your positive "let's give it a shot" attitude and charming personality.

I would also like to thank FORCE Technology, and especially Lars Vesth, for providing me with the opportunity to continuously advance my knowledge and career. Marcin Zajaczkowski and Harsh Jaiswal likewise deserve special recognition for their time spent assisting and testing the developed UAVs during my research in the Inspectrone Project.

A heartfelt thanks to my fellow Ph.D. student Rasmus Eckholdt Andersen for your commitment to both on-topic and off-topic discussions. It has been motivational to engage in our problem-solving sessions, and exciting traversing the ballast tanks together in the early project days before the pandemic.

During my studies, I have also been fortunate to work with NASA Jet Propulsion Laboratory (JPL), and would therefore like to thank Robert Hewitt for his highly impressive knowledge, guidance, and assistance in my research. This collaboration has created valuable connections, where Sarah Etter and David Millard deserve special recognition. They have added to the incredible experience at JPL and made many interesting late evenings possible, which have indeed opened my mind to dare mighty things.

Last but not least, I would like to extend an extra special thanks to Frida Agerskov for her patience during the more stressful periods of my research. This support cannot be understated, and I will forever be grateful for your support.

Dissertation Details

Thesis title:	Absolute Localization for Autonomous Robotic Inspection in confined spaces
PhD Student:	Rune Yding Brogaard
Supervisors:	Associate Prof. Evangelos Boukas, Technical University of Denmark Lars Vesth, Chief Operating Officer, FORCE Technology Professor Ole Ravn, Technical University of Denmark

The main body of the thesis consists of the following 7 articles.

Article A

Rasmus E. Andersen, Rune Y. Brogaard, and Evangelos Boukas. Autonomous robotic inspection for Remote Inspection Technique Systems: A review. *Field Robotics*, 3:69–96, 2023. doi: https://doi.org/10.55417/fr.2023002. Published.

Article B

Rune Y. Brogaard, Marcin Zajaczkowski, Luka Kovac, Ole Ravn, and Evangelos Boukas. Towards UAV-Based Absolute Hierarchical Localization in Confined Spaces. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 182–188, 2020. doi: 10.1109/ssrr50563.2020.9292616. Published.

Article C

Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. Absolute localisation in confined spaces using deep geometric features. *Electronics Letters*, 57(16):621–623, 2021. doi: https://doi.org/10.1049/ell2.12199. Published.

Article D

Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. GPU-accelerated Localization in Confined Spaces using Deep Geometric Features. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6, 2021. doi: 10.1109/ist50367.2021.9651425. Published.

Article E

Rune Y. Brogaard and Evangelos Boukas. Autonomous GPU-Based UAS for Inspection of Confined Spaces: Application to marine vessel classification. *Robotics and Autonomous Systems (RAS)*, 2023. Submitted.

Article F

Rune Y. Brogaard, Robert A. Hewitt, Sarah Etter, Arash Kalantari, and Evangelos Boukas. Absolute Localization in Feature-poor Industrial Confined Spaces. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023. Submitted.

Article G

Rune Y. Brogaard, Rasmus E. Andersen, Luka Kovac, Marcin Zajaczkowski, and Evangelos Boukas. Towards an Autonomous, Visual Inspection-aware 3D Exploration and Mapping System for Water Ballast Tanks of Marine Vessels. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6. Ieee, 2021. doi: 10.1109/ist50367.2021.9651476. Published.

Abbreviations

CAD	Computer-aided Design
CPU	Central Processing Unit
EKF	Extended Kalman Filter
FCGF	Fully Convolutional Geometric Features
FCU	Flight Controller Unit
FoV	Field-of-View
FPFH	Fast Point Feature Histograms
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPU	Graphics Processing Unit
IACS	International Association of Classification Societies
ICP	Iterative Closest Point
IMO	International Maritime Organization
IMU	Inertial Measurement Unit
KDE	Kernel Density Estimate
NBV	Next-Best-View
RANSAC	RANdom SAmple Consensus
RITS	Remote Inspection Technique Systems
RRT	Rapidly exploring Random Tree
SAE	Society of Automotive Engineers
SLAM	simultaneous localization and mapping
ToF	Time-of-Flight
UAV	Unmanned Aerial Vehicle
UN	United Nations
UWB	Ultra Wide Band
VIO	Visual-Inertial Odometry
WBT	Water Ballast Tank

Table of Contents

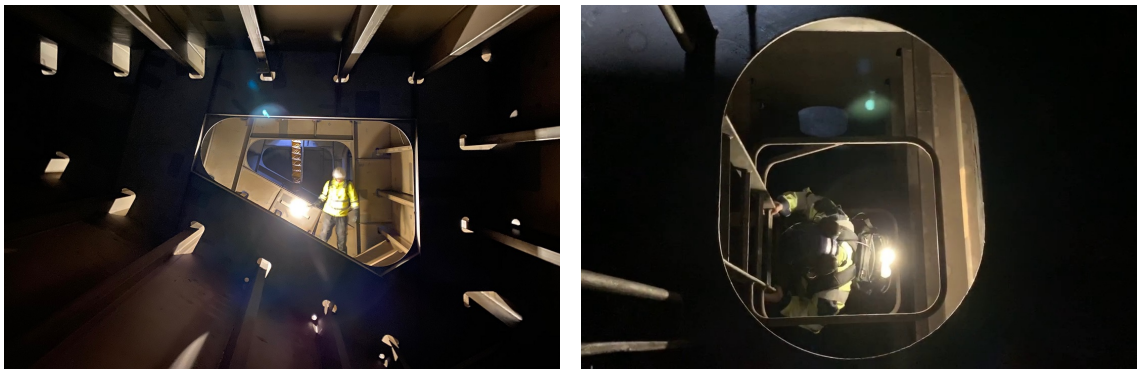
Chapter 1

Introduction

Inspection of industrial assets is a vital part of maintaining safe and reliable infrastructures across the world. Large-scale industrial inspections can be divided into a wide range of categories, where most share a common focus on determining the physical condition and integrity of a given asset. The asset can take many forms, where examples of such assets include power plants, refineries, storage tanks, wind turbines, buildings, ships, airplanes, and many more. Many of these assets include areas that are difficult to reach, explore, and navigate. Therefore, most of the data collection for inspection activities is currently carried out by human surveyors, who are able to climb and crawl into the odd locations necessary to evaluate the condition of the asset. Maneuvering around structures not intended for everyday human interactions poses an increased risk to surveyors, and entering confined spaces can be especially dangerous [8].

A confined space can be thought of as an enclosed box with the addition of a potential hazard [9]. The most deadly hazard present in confined spaces is the lack of oxygen [10], which cannot be detected by humans, but needs to be monitored by gas sensors mounted within the space and/or worn by the surveyor. The atmosphere in confined spaces can also be flammable, explosive, corrosive, or toxic. The dangers of the atmosphere can be caused by cargo, natural processes, such as the degradation of organic material, or work carried out within the space. For ballast tanks in marine vessels, such as the one shown in figure 1.1, a toxic atmosphere can arise from leakage of cargo tanks or pipelines, while a lack of oxygen can have several causes such as the corrosion process itself, the accumulation of rotting seaweed and other biological material in the empty tanks, as well as the intentional supply of inert gases to limit the corrosion process. The dangers of entering confined spaces are currently mitigated by employing different risk reduction strategies, but there is still an average fatality rate of 148 workers per year in the period 2011 to 2018 in the US alone[11]. Unfortunately, a third of these casualties are caused by other personnel rushing into the space to rescue the collapsed person. From a safety perspective, it is clear that solutions are needed to reduce the fatalities in enclosed spaces.

Using robotic systems to replace humans for dangerous operations is one way to reduce this number of casualties. Likewise, robotic solutions can also be used in areas of dull and dirty work to free up the workforce to tackle more complex tasks. Additionally, a robotic inspection solution will be able to perform regular and repeatable inspections that could furthermore provide trend line data analysis and metrics by which the asset owner can

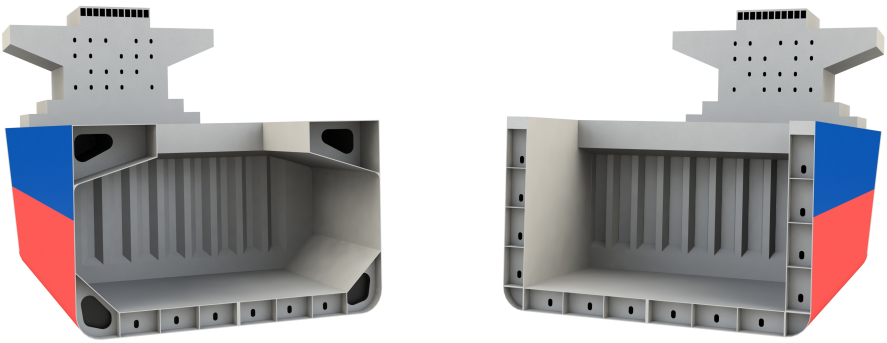


(a) *(b)*

Figure 1.1. An example confined space inside a marine vessel. Figure (a) captures the inside of a topside water ballast tank of a single hull bulk carrier. Figure (b) shows the author entering the deeper levels of the side water ballast tanks of a double hull oil tanker.

efficiently plan and cost-manage future repairs. The research covered in this dissertation is mainly focused on the inspection of ballast tanks for marine vessels, however, experiments were also carried out on a mock-up model of a chemical distillation column using a legged robot from Unitree (see article F).

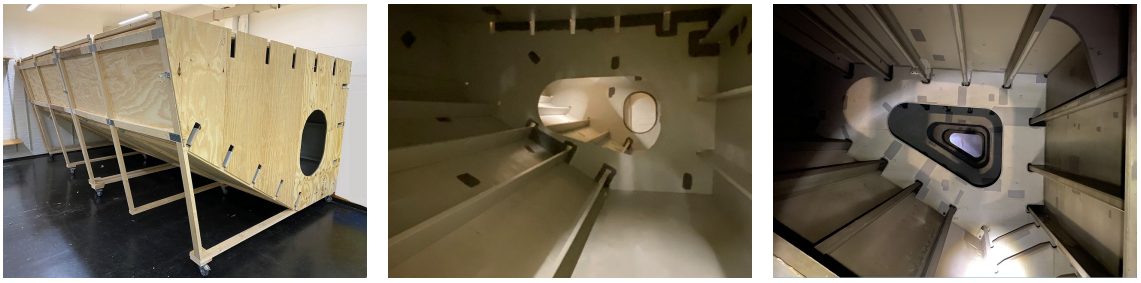
Marine vessels are a critical part of the modern world infrastructure, since 70-80 % of goods are transported at some point by ships[12]. Furthermore, the environmental impact of a shipwreck can be severe, and therefore regulations are established to avoid catastrophic structural failure of vessels. The surveys of the vessels are carried out at fixed intervals, and part of these focus on performing visual inspection of the water ballast tanks. The water ballast tanks serve the purpose of maintaining good alignment and stability of the vessel under different cargo types and cargo hold filling levels. To maintain the desired alignment of the vessel in the sea, the ballast tanks can be flooded with various amounts of water at different locations on the ship. However, the ballast tanks are mostly empty when the ship is transporting cargo, which can be expected to be the majority of its useful life. The water ballast tanks on most commercial grade marine vessels are generally located in the bottom and near the sides of the ship hull.



(a) *(b)*

Figure 1.2. A cross-section of single skin and a double skin vessel. Ballast tanks are placed either on the side, directly underneath the deck (topside tanks), or around the cargo holds (double hull side ballast tanks).

Figure 1.2 illustrates the ballast tanks of (a) a single-hull bulk carrier and (b) a double-hull tanker. Most of the experiments in this dissertation are conducted on a mock-up model of a top-side water ballast tank of the single-hull tanker NORD SUMMIT (D/S Norden). The mock-up model described and used for the experiments in the articles included in this dissertation, is shown in Figure 1.3. The mock-up model is designed at a scale of approx. 2:3, with the exception of the man-ways, which are kept at a 1:1 scale to allow a Unmanned Aerial Vehicle (UAV) to enter the model. The man-ways have a width of 0.6 m and a height of 0.8 m. The total length of the two compartments in the ballast tank is 5.2 meters with a maximum width of 3 meters and a maximum height of 1.75 meters.



(a) Outside mock-up model ***(b)*** Inside mock-up model ***(c)*** Inside real ballast tank

Figure 1.3. The mock-up model compared to the inside of a real topside water ballast tank from a marine vessel. From (b) and (c) it can be seen that the mock-up model has a good resemblance with the real world scenario.

1.1 Problem statement

When inspecting or navigating within confined spaces, localization is a major concern for both humans and robots[13]. Most inspection results are invalid without a proper position tag added to the data point. The severity of a detected defect can depend greatly on where it is located. For example, the exact location of a corrosion spot can determine whether an asset needs to be immediately shut down or continue its work, with further more in-depth investigations scheduled at a later date. Localization in confined spaces poses a range of challenges. Using a Global Navigation Satellite System (GNSS), such as Global Positioning System (GPS) and Galileo, to provide the pose estimates would seem as an obvious choice. However, the lack of radio transparency in many enclosed spaces makes it an unviable solution for most cases. A similar approach to the GNSS method is the installation of local Ultra Wide Band (UWB) based satellites inside the space, which was also investigated by [14, 15, 16]. This satellite-based system can provide a reasonable pose estimate for factory floors and storage facilities [17, 18], but the requirement of mounting localization hardware inside the space makes them unusable for our described use cases. Mounting hardware inside the space violates the concept of limiting human entry into the confined space. In addition, installing satellite hardware during construction is also often not an option due to both the cost and the —hostile to electronics— environment of the confined space. An alternative but similar method, which simplifies the UWB concept, would be to install fiducial markers (eg: the popular ArUco markers) at known fixed locations within the confined space. The markers can then be detected and interpreted by a camera on the robot to provide a position estimate. This has been investigated in [19, 20, 21], and can provide a good estimate of the absolute position. However, it still requires an installation

of the markers and an accurate measurement of their positions, which adds cost and also the requirement to keep the tag clean and visible before a robot-based inspection is conducted. It is also important to consider that there exist approximately 54,000 vessels[22] in operation and any solution that requires predefined landmarks (after construction) is almost unfeasible. To avoid installing known landmarks into the environment, we could use Visual (Inertial) Odometry as our position estimate, which builds upon the idea of extracting and tracking visual features from a camera image stream. Known methods include SVO[23], DSO[24] ROVIO[25], and VINS-mono [26]. Although these methods are generally considered to provide a good and robust relative pose estimate, they still require sufficient lighting conditions, and their performance is limited by the number of available features. Many industrial spaces have sections that are featureless and increase the risk of unbounded errors, i.e. drift, for these purely visual-based tracking systems.

To solve the lack of visible trackable features, one could use the 3D structure of the confined space to provide an absolute position estimate of the robot. Extraction of the structure around the robot can be achieved in multiple ways, but is often done by either stereo vision or LIDAR based solutions. Stereo vision can create a depth image or a point cloud of the field of view by detecting visual features from each camera image and then finding the corresponding features between the two images. However, as described earlier, visual features are not abundant in industrial confined spaces, and the lack of features in the images of the stereo camera can quickly reduce the available points in the generated point cloud. Therefore, structured light in the form of randomly arranged dots, or similar, can be shined onto the surface to create visual features on otherwise featureless surfaces. The point clouds generated in article B used this type of structured light depth camera.

3D spinning LIDARs can, on the other hand, provide accurate 3D point clouds over 360 degrees without the need for a feature-rich surface. The LIDARS are frequently used in self-driving cars and other wheeled robots and can detect reflective surfaces up to several hundred meters away. However, this type of LIDAR sensor has several disadvantages; it can be too heavy for small rotor-based aerial systems, it has a higher rate of failure than non-mechanical solutions and it generally has a higher energy consumption than non-spinning devices. Therefore, smaller Time-of-Flight (ToF)-based cameras might be the only option for smaller UAVs. In the project supporting this research, Inspectrone, the available spinning LIDARs were considered too heavy for the aerial robot, and a mixture of stereo cameras and ToF cameras were used to provide 3D point clouds of the environment. By choosing these cameras, the generated point cloud can only cover a limited area determined by the field of view of the camera. The limited point cloud coverage further adds difficulty in determining where the point cloud was generated, due to the ambiguity of the environment, as was also presented in article C, D, F.

1.2 Research purpose

The main purpose of this Ph.D. project is to develop an absolute localization system to be able to perform autonomous robotic inspection of confined spaces. The developed system will be part of the navigation system of the Inspectrone project, which aims to automate the inspection of confined spaces in marine vessels, by using a small UAV. For most of these vessels, as well as other industrial assets, the Computer-aided Design (CAD) files for

their structure are available, or can be acquired and can, as such, be utilized as an aid for the localization system. It is therefore worth investigating the possibility of generating absolute pose estimates by registering structural scans from the interior of the space to the CAD model of the vessel.

In rare cases where a CAD drawing of the confined space is not available, an exploration of the environment should be investigated with a focus on collecting inspection data from it.

The objectives of this PhD research can therefore be summarized as follows:

1. **Provide an absolute localization system for known confined spaces tailored to data availability.**
   The approach for absolute localization in confined spaces will vary depending on the available data. If no data are available, exploring the environment is a necessity. For older industrial spaces, there are often simple drawings of the confined space that can be used for a high-level form of absolute localization. Similarly, most modern infrastructure assets contain CAD models that can be utilized as a reliable prior map for absolute localization.
2. **Provide absolute localization in visual featureless or visually degraded confined environments, specifically by providing/employing a well formed uncertainty.**
   Industrial spaces are often constructed using steel or stainless steel, which has very few visual features, and often contains a high degree of symmetry. This means that the system needs to be able to use the 3D structure of the environment to provide an absolute pose estimate, and at the same time be able to provide a reasonable localization uncertainty for the ambiguous environment.
3. **Create absolute localization pipelines that are specifically designed for online operation on small form-factor robots.**
   Size limitations in the access ways for confined spaces induce the need for small and agile mobile robots to traverse the environment. Small robots have limited computational and power capabilities, leading to a need for a selection of pipeline architectures that can be either parallelizable on dedicated co-processing units or accelerated in a similar way to maintain real-time capabilities.

If these goals can be achieved, it can lead to safer and more repeatable inspection of confined spaces. This improvement in industrial inspections also has the potential to reduce costs and gain access to areas that previously would have been considered too dangerous for human entry.

1.3 Summary of contributions

In this section, the contributions of the dissertation and the motivations behind each publication are explored. Section 1.3.1 provides a list of the individual articles and their significant contributions.

The first contribution in this dissertation is a literature review of current state-of-the-art research for autonomous inspection of marine vessels. This review is focused on the maritime industry and identifies the autonomy levels of the different inspection systems.

On the one hand, the target is on identifying how efficiently the systems traverse the different environments of the vessels, whether this is outside or inside the ship. On the other hand, the review of systems for automatic detection and evaluation of the collected inspection data was performed. In addition, it provides potential research areas in the field of maritime inspections, which can surpass current state-of-the-art methods.

The second contribution in this dissertation is the creation of an absolute localization system, developed to traverse known confined spaces using custom high-level landmarks. High-level landmark extraction limits the requirements of the available map of the space with the benefit of maintaining a low memory footprint of the onboard computer system. The complete system also has the capability to estimate the pose of defects, as well as a hierarchical structure of the landmarks for simpler navigation. This contribution meets the need in the maritime industry for a simple localization system without the need for detailed CAD drawings, and thus is mostly suited for the aged and soon to be scrapped vessels.

The third topic of contributions tackles the need for more generalizable structural landmarks for absolute localization in confined spaces. Using a range of 3D geometric feature descriptors, different localization pipelines are investigated, focusing on localizing within the water ballast tanks of marine vessels. GPU acceleration is introduced in the form of novel Deep Learning descriptors, and Graphics Processing Unit (GPU)-based KNN search is employed to increase the frequency of absolute pose estimates. The contribution of GPU-accelerated localization pipelines enables a faster method of pose estimation compared to the standard CPU-based alternative.

The fourth area of contribution addresses the requirements for providing an uncertainty estimate for absolute localization in confined spaces with a high level of similarity. This contribution takes advantage of a GPU accelerated Stien-ICP based uncertainty estimation method to aid in localizing, especially in environments with a high level of similarity. Two systems are designed, where the first one adds the uncertainty estimation to a deep learning based localization pipeline. The second system utilizes custom-defined landmarks and incorporates the inherent registration capabilities of the Stein-ICP uncertainty method, to provide an absolute pose with uncertainty within the cylindrical structure of a distillation column.

The fifth and final contribution of this dissertation disregards the previous assumption of a prior map and investigates the possibility of an exploration-based autonomous inspection system. This contribution introduces a defect-aware exploration of confined spaces, where the goal of the exploration planner is, additionally to maximizing the known areas, to obtain a large amount of data of detected defects. The collection of a large amount of photos of potential defects enables the possibility of an automatic evaluation system for further research.

1.3.1 Publications

Journal articles

Article A

Rasmus E. Andersen, Rune Y. Brogaard, and Evangelos Boukas. Autonomous robotic inspection for Remote Inspection Technique Systems: A review. *Field Robotics*, 3:69–96, 2023. doi: https://doi.org/10.55417/fr.2023002. Published.

A literature review of current state-of-the-art marine vessel inspections, which introduces a metric to score the autonomy of the systems. The main contributions are as follows:

- A scoring metric to evaluate the current state of the art with regard to the industry standards as set by the classification societies, specifically with regard to fully autonomous inspection.
- Detailed overview of accessible research and technology for autonomous robot navigation and defect detection in marine vessels.
- Pinpointing future research areas that will increase the level of autonomy of inspection UAVs.

Article C

Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. Absolute localisation in confined spaces using deep geometric features. *Electronics Letters*, 57(16): 621–623, 2021. doi: https://doi.org/10.1049/ell2.12199. Published.

This journal article uses novel machine learning-based feature descriptors for 3D point cloud registration, in combination with Visual Inertial Odometry, for 3D absolute pose estimation. The main contributions are as follows:

- Development of an absolute localization system using deep learning-based 3D feature descriptors for robots operating in confined spaces.
- Integration of a fast and robust registration algorithm for 3D registration in confined spaces under the presence of a large number of outliers.
- Comparison of well-known designed 3D features (FPFH) with deep-learning equivalent.

Article E

Rune Y. Brogaard and Evangelos Boukas. Autonomous GPU-Based UAS for Inspection of Confined Spaces: Application to marine vessel classification. *Robotics and Autonomous Systems (RAS)*, 2023. Under review.

This journal article presents and demonstrates a real-time inspection system for collecting 3D structural and visual data, from a series of inspection points within a prior map of a confined space. The main contributions are as follows:

- Online GPU based 3D localization system using Deep Learning features.
- Integration of a novel uncertainty estimation of a 3D-3D registration in a localization filter.
- Development of an autonomous inspection system for known feature-poor confined spaces.
- Showcasing a full inspection pipeline onboard an autonomous aerial robot.

Peer-Reviewed Conference Articles

Article B

Rune Y. Brogaard, Marcin Zajaczkowski, Luka Kovac, Ole Ravn, and Evangelos Boukas. Towards UAV-Based Absolute Hierarchical Localization in Confined Spaces. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 182–188, 2020. doi: 10.1109/ssrr50563.2020.9292616. Published.

This conference article creates an absolute and hierarchical localization system employing a combination of Visual Inertial Odometry and detection of existing known structural 3D landmarks. The main contributions are as follows:

- Introduction of a simple and hierarchical landmark-based extraction system for water ballast tanks.
- Creation and testing of a custom absolute location system based on landmarks to inspect the water ballast tanks of marine vessels.
- Detection and localization of simulated defects in marine vessels.

Article D

Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. GPU-accelerated Localization in Confined Spaces using Deep Geometric Features. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6, 2021. doi: 10.1109/ist50367.2021.9651425. Published.

This conference article uses a Graphic Processing Unit(GPU) to enable the possibility of deep learning-based feature descriptors for near real-time absolute localization in confined spaces. The main contributions are as follows:

- Introduction of a GPU accelerated localization pipeline for confined spaces.
- Performance comparison of state-of-the-art deep learning-based feature descriptors in an environment with a high level of self-similarity.
- Deployment of a robust localization system, tested in a dataset from a real environment.

Article G

Rune Y. Brogaard, Rasmus E. Andersen, Luka Kovac, Marcin Zajaczkowski, and Evangelos Boukas. Towards an Autonomous, Visual Inspection-aware 3D Exploration and Mapping System for Water Ballast Tanks of Marine Vessels. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6. Ieee, 2021. doi: 10.1109/ist50367.2021.9651476. Published.

This conference article extends a receding horizon "next-best-view" exploration planner to increase the exploration towards detected defects. The main contributions are as follows:

- Defect-aware exploration of confined spaces.
- Novel objective function in a well-known exploration technique including defect areas for an "inspection-aware" exploration.
- Development of a photorealistic ballast tank in a real-time simulator.

Article F

Rune Y. Brogaard, Robert A. Hewitt, Sarah Etter, Arash Kalantari, and Evangelos Boukas. Absolute Localization in Feature-poor Industrial Confined Spaces. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023. Under review.

This conference article presents an uncertainty-aware absolute localization system for highly ambiguous spaces, using visual inertial odometry and GPU-based point cloud registrations for limited field of view sensors. The main contributions are as follows:

- Incorporating the uncertainty-aware 3D registration algorithm, Stein-ICP, on an online robotic system for EKF-based absolute localization.
- The improvement of Stein-ICP based estimation by using the Kernel Density Estimation distribution evaluation.
- Taking advantage of prior information about the environment and robot to both speed up Stein-ICP (by sampling based on state uncertainty) and to reduce scene ambiguity by only registering local views to unique elements of the environment.

Chapter 2

State of the Art

Autonomous inspection of marine vessels has received increasing interest as the shipping industry grows. Many areas of the vessels are still manually inspected, but research has been conducted on how to automate a range of these inspections. The purpose of this chapter is to provide a review of the literature on the current state-of-the-art in automated inspection. The following article will investigate two aspects of the procedure for automated inspection, namely, expertise and engineering. The expertise part is focused on the automatic evaluation of the collected inspection data, while the engineering part is focused on in which ways the robotic system can get to the desired inspection point, and with what level of autonomy. These capabilities will be related to the required specifications from the classification societies, i.e. DNV GL and Lloyd's register, which are some of the most important stakeholders among the end users. One of the objectives of classification societies is to verify the structural integrity of ships to both government agencies and insurance companies. Most structural integrity inspections are currently performed by human surveyors, where the amount of corrosion is evaluated by each surveyor following the guidelines provided by the classification society. It is clear that this is subject to each surveyor's judgement, and an automatic evaluation solution could potentially avoid human discrepancies between surveyors and could also speed up the evaluation process. Therefore, the available expertise aspect of the automatic inspection process is evaluated in the article using a set of metrics, such as the system's ability to detect defects, classification, and evaluation.

There is also a need to avoid human entry into the confined spaces onboard the vessel. Therefore, a robotic solution is required, hence the engineering part of the article. Different kinds of robotic systems have been tested by researchers in the investigated literature, with a wide range of automation levels depending on the operation area. The autonomy level is quantified using a taxonomy inspired by the Society of Automotive Engineers (SAE) taxonomy [27]. The SAE taxonomy was originally intended for motor vehicles on the roadway system, however, the same classification can within reason be applied to any autonomous vehicle. Thus, the gradual removal of the operator is similar for both grounded vehicles and aerial vehicles, ultimately progressing to full autonomy without the influence of a human operator. It is important to note that the metric does not evaluate the robustness or performance of each system. The accuracy, precision, or how they handle disturbances could vary widely, both in and across categories. However, it provides an overview of the research-based solutions currently available for the shipping industry and the areas that require further investigation.

Using different sets of metrics for the engineering and expertise aspects of an autonomous inspection solution, the included article reviews and quantifies the research available in the shipping industry. The findings conclude that currently there is no fully autonomous SAE level 5 solution for confined space inspection. Some methods in level 4 exist for exploring cave and mine environments, but have not yet reached inspection of marine vessels. Similarly, the lower level of autonomy still has humans in the loop for conducting the inspection, which is especially true for inspection of the water ballast tanks of marine vessels. Little literature exists in this area, furthermore revealing that confined space inspection of marine vessels is an area where more research is needed. The review, furthermore, indicates that the use of a deep learning architecture is suitable for the data evaluation part, and a flexible robotic solution such as an aerial vehicle could be an effective inspection solution, as illustrated in Figure 2.1. The combination of these two approaches was tested in a simulated ballast tank in **article G**.

(a) UAV in the ballast tank ***(b)*** Corrosion detection using Deep Learning

Figure 2.1. Autonomous inspection of water ballast tank using an UAV in the simulation environment AirSim. The UAV used in the simulation is shown in (a), where its camera point of view, with the detected corrosion, is shown in (b).

Article A

Autonomous robotic inspection for Remote Inspection Technique Systems: A review

Rasmus Eckholdt Andersen[1], Rune Y. Brogaard[2], Evangelos Boukas[1]

[1]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark
[2]FORCE Technology, Park Allé 345, Brøndby, Denmark

Abstract:

Due to the harsh environment and heavy use modern marine vessels are subjected to, they are required to undergo periodic inspections to determine their current condition. The use of autonomous remote inspection systems can alleviate some of the dangers and shortcomings associated with manual inspection. While there has been research in the use of robotic platforms, none of the works in the literature evaluates the current state of the art with respect to the specifications of the classification societies, who are the most important stakeholders among the end users. The aim of this paper is to provide an overview of the existing literature and evaluate the works individually in collaboration with classification societies. The papers included in this review are either directly developed for, or have properties potentially transferable to the marine vessel inspection process. To structure the review, an expertise-engineering separation is proposed based on the contributions of the individual paper. This separation shows both which part of the inspection process has received the most attention as well as where the shortcomings of each approach lay. The findings in this review clearly indicate that while there are promising approaches, according to our metrics, there is still a gap between the classification societies' requirements and the state of the art.

Our results indicate that, even though there is a lot of quality work in the literature, there is a lack of integrated development activities that achieve a level of completeness sufficient for the classification societies to confidently use them.

A.1 Introduction

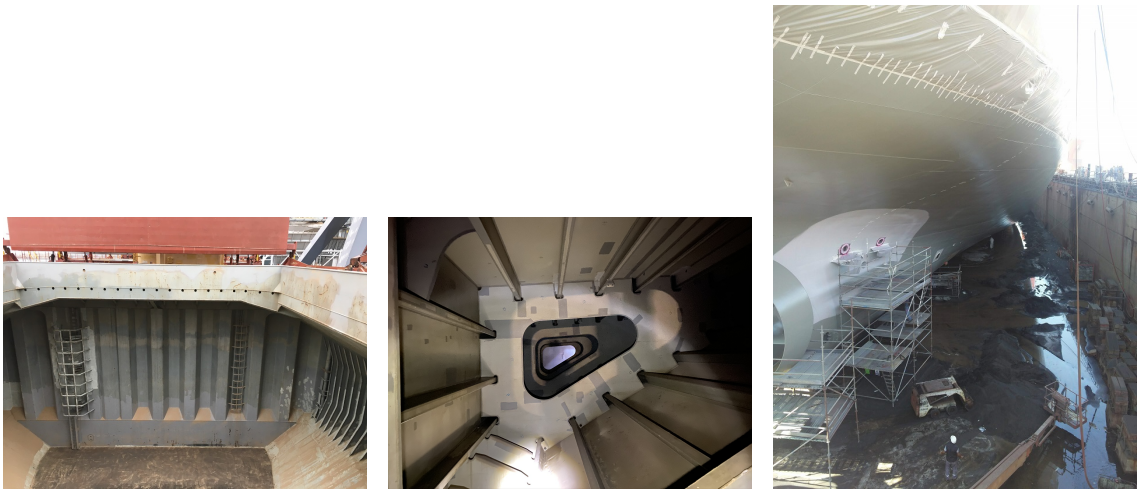
One of the most efficient ways of transportation is by sea, which, according to the United Nations (UN) [28], constitutes over 80% merchandise trade by volume. It is critical for the environment, that marine vessels —especially the large vessels which transport cargo— are safe to operate to minimize the risk of contamination e.g. oil spills or more abrupt catastrophes such as explosions.

(a) *(b)*

Figure A.1. Examples of marine vessels (a) during operation and (b) during maintenance in a dock.

The International Maritime Organization (IMO) is a specialized agency of the UN and the authority in setting standards for safe shipping. Its main contribution is a universally adopted regulatory framework ensuring that ship owners cannot sacrifice safety for increased market advantages. The level playing field set by the IMO covers all aspects of shipping including construction and maintenance of marine vessels [29].

The regulations dictate that cargo vessels have to undergo periodic inspections where, among other tasks, the condition of the vessel is assessed. The main goal of these inspections is to find any defects present on the vessel that may reduce its structural integrity and thus pose a risk of failure during operation. Examples of the most important areas that undergo inspections are shown in figure A.2. The actual inspection is carried out via a collaboration between the vessel owner and a classification society or a class-certified surveyor that enforces the regulations set by IMO. It is up to the individual classification society how they ensure that the regulations set by the IMO are adhered to and it is, thus, up to them what technologies and assisting tools they deem suitable for use in the assessment. For this reason, 12 classification societies have formed the non-governmental International Association of Classification Societies (IACS) [30] that provides technical support and guidance on the unified regulations set by the IMO. Additionally, they provide support when amendments and changes to the existing regulations occur or new regulations are added.



(a) *(b)* *(c)*

Figure A.2. Examples of areas on a modern marine vessel that have to be inspected (a) cargo hold (b) ballast tank (specifically top-side tank) and (c) the outside hull of a vessel.

One of the emerging technologies that classification societies are adopting at an increasing rate is the use of drones (an example of a drone flying in a ballast tank is shown in figure A.3). They serve the purpose of removing human surveyors from the hazardous and unfriendly environments present on a vessel during inspections while still allowing the acquisition of sensory information used in the assessment.

Figure A.3. Example of a drone flying in a ballast tank. If equipped with the right sensory equipment accompanied by a high level of autonomy, the drone can replace human surveyors in the hazardous environments present in the ballast tanks of modern marine vessels.

With the recent increased availability of viable drone solutions and the advances in image processing and recognition, the works in the literature trying to automate the inspection process have also increased. The existing literature tends to either focus on detecting defects [31, 32, 33, 34], or to provide drones capable of reaching the areas that are difficult to access while balancing the trade-offs of equipment payload, drone size, operation time, etc [35, 36].

The IACS has defined some recommendations for how these Remote Inspection Technique Systems (RITS) should operate, however, the general interpretation of it is, that any RITS has to be able to represent the same quality of information as a surveyor being physically present would be able to acquire [30]. Thus there are no clear definitions or requirements for the drone, as long as the assessment does not suffer in quality. Thus, a complete/perfect RITS should be capable of delivering the same level of quality inspection as a human surveyor, with all the added benefits of task automation such as easier standardization, repeatability, increased precision, etc. Specifically, the repeatability is of great importance since a vessel (shown in figure A.1) can contain more than 600 000 m^2 of steel that has to be visually inspected within a 1-2 meters observation distance. It is nearly impossible for a human surveyor to maintain focus for the time required to inspect such an amount of steel, resulting in varying and subjective inspections [37].

Though it is clear that some progress has been made, it is unclear how close to a complete RITS any single work or collection of works has come, since there exists a general lack of evaluation concerning the regulations as a whole. Thereby, it is hard to quantify how close the community has come to a realistic solution that the classification societies can use.

A previous survey [38] evaluated the technological advances, particularly within marine vessel inspection. They present defect detection as the detection of cracks and corrosion and robotic platforms suitable for marine vessel inspection. While they provide a comprehensive list of the state of the art, they do not address all of the criteria for a solution to be viable for the classification societies, such as being able to quantify the defects or detecting the type of corrosion.

This paper strives to fill this hole and, in collaboration with a classification society (Lloyd's Register [39]), evaluate the current status of how close the state of the art is to a fully functioning RITS. This is done by separately investigating a) the expertise and b) the engineering challenges present when capturing information to apply that expertise. Specifically, this paper is interested in identifying the existing works in the literature that encapsulate both the expertise and the engineering as shown in figure A.4. For this paper, expertise is defined as the knowledge and the associated processing ability to assess the condition of a vessel given a selection of sensory information. The engineering aspect concerns the acquisition of the information while raising the autonomy level higher than that of a fully teleoperated drone solution. Thus, engineering encapsulates the physical hardware and control software needed not only to autonomously navigate the relevant inspection areas of the marine vessel but to additionally ensure that the requirements for applying the expertise can be fulfilled with sufficient sensory information.

In section A.2, the approach employed to evaluate the progress of the state of the art is presented. The metrics defined in section A.2 will be used to score the existing literature within what has been defined as expertise in section A.3 and engineering in section A.4.

A.2 Metrics definition

The assessment parameters of the marine vessels are derived from the type of materials used for their construction and are defined by the classification societies. An example is that the condition of the paint (coating) is assessed to be either good, fair, or poor, if there

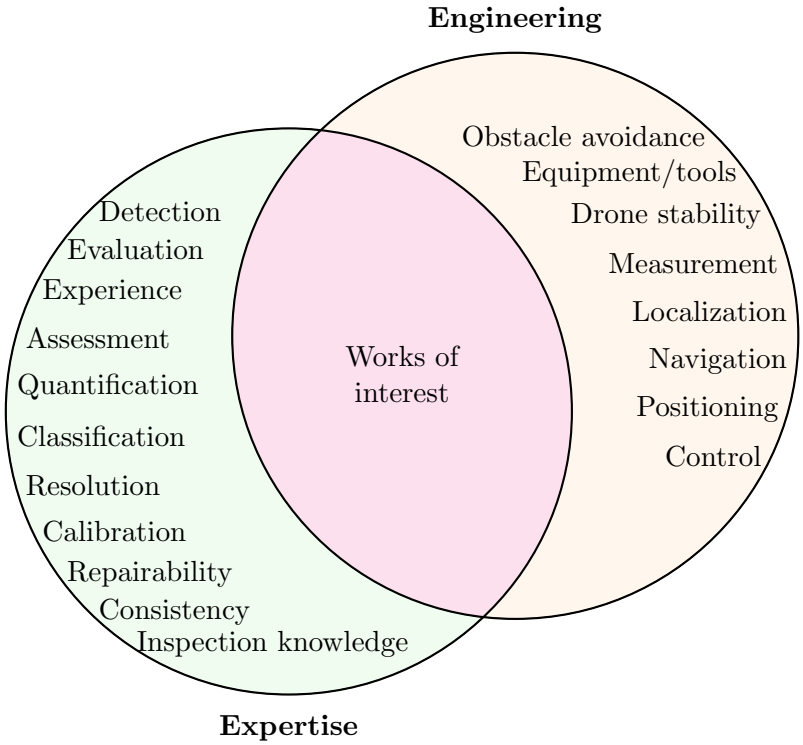


Figure A.4. The role of the expert surveyors is to supply their vessel inspection knowledge. However, this knowledge has to be based on sensory information that binds said knowledge to a specific physical context. Thus, any solution attempting to automate a process will have to both capture expertise and gather information. In the context of marine vessel inspection, the knowledge relates to defect detection, classification, and repairability. Engineering relates to supplying the necessary information and navigating the vessel. Thus, in the current work, the literary works of interest are those which tackle the overlap between these two concepts.

is less than 3%, between 3% and 20%, or more than 20% corrosion, respectively. Similarly, cracks are quantified by their location and physical sizes, such as length and width.

The main job of a surveyor is to document the type of defect, its location, and severity — which in this paper are collectively referred to as expertise since it encapsulates the knowledge and training of the surveyor. Similarly, the engineering aspect is the technology needed to support the expertise. Analogously, for a manual inspection process, this could be the modular scaffolding or the mobile lights required to inspect all areas of the vessel. The following section is a description of how the existing literature is evaluated. The goal of this work is to quantify the development by the community so that it is possible to identify solutions that solve either the engineering challenges related to using autonomous drones in confined spaces and/or that try to capture the expertise of human surveyors by evaluating sensory information to assess the condition of marine vessels.

A.2.1 Expertise

Though there exist defined criteria with regards to the condition of a vessel, few papers in the literature compare against them and, instead, tend to use standard image processing metrics such as confusion matrices and intersection over union. Even though such an approach evaluates the performance of the detection algorithm —and is, thus, important— it does not reveal whether any progress is being made towards the actual goal of assessing

the condition of the vessel in relation to the unified requirements. In this paper, the existing literature is evaluated on six different parameters that either represent parts of the survey used to document the vessel condition or the level of autonomy that the literature is trying to introduce into the assessment process.

Seen from the point of the surveyor, some of the tasks related to vessel inspection can be more critical than others. For instance, it may be of higher importance to the surveyor that the autonomous evaluation can detect the type of defect, than that it is able to determine its location. For this reason, the importance of each evaluation metric is weighted by a classification society (Lloyds Register). The weight ranges from 1-4 with 4 being most important and 1 being of lesser importance.

The final score of existing literature will then be given by the following formula:

$$Score = \frac{w_i \cdot v_i}{\sum_i w_i} \tag{A.1}$$

where w_i denotes the weight given by the classification society and v_i is the score given for the metric i. The purpose of this evaluation is to quantify the minimum level of expertise captured by any method before it can realistically be considered by the classification societies. The weights given by the classification societies are normalized to be within the same interval as the scoring of the metrics used. For instance, if there is a metric that the classification societies consider essential, but none of the methods score high on this metric, the overall score of the method should be lowered since the presented method would not be of interest to the classification societies. The metrics used are the following, and a definition of the metrics can be found in the related section:

1. Defect detection (section A.2.1)
2. Defect location (section A.2.1)
3. Defect quantification (section A.2.1)
4. Autonomous evaluation (section A.2.1)
5. Flexibility (section A.2.1)

To increase reproducibility, table A.1 shows some general examples of what each scoring means.

Defect detection

The ability to detect defects based on sensory inputs is a prerequisite to the works evaluated in this paper. The necessity of this parameter is to sort out approaches that only process or otherwise transform sensory information before any real assessment or evaluation is done. Since this metric is a prerequisite to be considered at all in this evaluation, it has been left out of the final scoring as it only adds a constant bias to all the works evaluated.

Example	Score
Defect type	
A work that performs binary classification (i.e. only concerns the presence of defects/corrosion or otherwise abnormalities).	0.1
A work that actively includes one of the corrosion types stated by the classification society.	0.5
A work that actively includes all of the corrosion types stated by the classification society.	1.0
Defect location	
A work that classifies on a whole image basis without respecting spatial location in any form.	0.0
A work that provides bounding boxes or otherwise proposes some weak form of localization (e.g. through recursive detection) and is strictly bounded to the image plane.	0.5
A work that provides low-level positioning of the defect with respect to some extrinsic reference frame.	1.0
Defect quantification	
A work that does not quantify any detection.	0.0
A work that is able to quantify the detected defects.	0.5
A work that is able to quantify the detected defects such that they can be compared against the existing metrics used by the classification society.	1.0
Autonomous evaluation	
A work that relies heavily on manual operation to function, e.g. a surveyor has to traverse the area with some equipment or remove samples from the vessel to analyze off-site.	0.0
A work that functions partially automates the evaluation process but still relies on some human interaction, e.g. for result interpretation.	0.5
A work that is able to carry out the entire evaluation process without any prior existing system knowledge.	1.0
Flexibility	
A work that is dedicated to inspecting a single specific element of the marine vessel.	0.0
A work that is dedicated to inspecting parts of a single area of the marine vessel (e.g. parts of the hull)	0.5
A work that is able to be deployed in one, or more, areas of the marine vessel that requires inspection.	1.0

Table A.1. Examples of how a score is assigned.

Defect type

In order to identify the severity, the type of defect is important since the maintenance process changes depending on how the vessels' structural integrity is affected. In the standards set by IACS [30], there are 4 main defect types; Cracking, Deformation, Coating Breakdown, and Corrosion. It is these types of defects that a surveyor inspecting the structural integrity of a vessel is expected to document and base a final vessel classification on. The underlying interest of the defect type is determining the likelihood of failure and propagation (e.g. a growing crack) which vary among the four types. Thus, it is fundamental that in order to achieve a complete, or close to complete, the autonomous inspection process, any solution or algorithm has to be able to identify the type of defect such that it can be documented in a vessel condition report. Since there are multiple types of defects to be detected, the score has to reflect how many of the necessary defects can be identified. The classification society weighted this as 4, with the reasoning that the type of defect is important to estimate the severity of the defect and thereby the extent of necessary repairs.

Defect location

In order to efficiently plan any work for maintenance, the location of any defect is required. In this paper, the location of the defect is not necessarily with respect to the vessel itself,

but rather just to a fixed frame. The fixed frame can either be the physical location in relation to the vessel or with respect to some other frame such as a sensor frame. Depending on the area of the vessel being inspected, the location can be of higher or lower importance. Generally, the primary structures (e.g. traverse bulkhead, longitudinal bulkhead) are of more importance than secondary structures (e.g. stiffeners, sides, deck, bottom). The surveyor documentation requires that the defects are located with respect to the vessel structure, rather than an arbitrary relative frame (e.g. a camera sensor). This has to be reflected in a score. The weight given to this metric by the classification society is 3, as the location of the defect is important but not critical, since larger areas could be scanned in smaller increments and, thereby, still, reduce the total area requiring subsequent manual inspection.

Defect quantification

A critical part of documenting the condition of the vessel under assessment is to quantify the defects. In some cases, like cracks, this requires physical measurements in terms of length and width, but for other cases, the quantification is more ambiguous. Such a case is corrosion which is measured in the percentage of *the area under consideration* [39]. For visual sensory inputs like imagery, a scale readout is required. If this is not provided, the physical size of the defects is subject to perspective distortions. Similarly, imagery data must be accompanied by some form of image quality indicator along with a calibration procedure that ensures the quality of the sensory information. Similar to defect type, there are multiple ways of quantifying defects, and thus the score has to reflect how many ways any solution or algorithm can quantify defects. With similar reasoning as for defect location, the metric is given a weight equal to 2. Specifically, the quantification of the defects can in some scenarios be done manually without losing the advantages of an otherwise autonomous solution.

Autonomous evaluation

Some of the proposed solutions in the literature may attempt to only assist the existing manual inspection process. This can for instance be achieved by evaluating gathered data offline or by representing it intuitively but still relying on manual inspection. The autonomous evaluation score is, thus, an encapsulation of how many manual processes are required to be performed for the autonomous inspection. The classification society weighted this metric with a value of 1 since the goal of using autonomous solutions is to remove the human surveyor from hazardous environments. Thus, if the human surveyor is required to be present in the evaluation process, the benefits of performing autonomous evaluations diminish.

Flexibility

Marine vessels contain diverse types of areas that require inspection for defects, ranging from large open spaces such as cargo holds, to small confined spaces like ballast tanks. The need for this metric is to identify the works in the literature that focus on specific areas and those that can perform inspections in multiple areas of the vessel. An example of this distinction is a solution that only works in water on the outside hull of the vessel and a solution that can be used in cargo holds vessel hulls and ballast tanks. Flexibility was

given a weight equal to 4 by the classification society with the reasoning that a significant motivation for using a RITS is the ability to reach otherwise hard-to-access areas. Such areas are present all over the vessel and being able to use the same solution for multiple areas of the vessel is of great importance.

A.2.2 Engineering

The engineering aspect of performing inspections is the process of acquiring the information required to utilize or apply the expertise. If the inspection is performed on the vessel hull while in water, the engineering aspect is to create a solution that can navigate in water. Since expertise and engineering are separate, this category will also include all those solutions that utilize different forms of drones in a teleoperation setting and still rely on human surveyor expertise for the actual assessment. A pure teleoperated solution still solves many of the challenges present in the current way of performing inspection, such as removing the human surveyor from the hostile environment present in ballast tanks. However, a significant part of the engineering challenge is to be able to traverse the vessel and since this has to include some form of autonomy, the engineering score used to evaluate the existing literature is derived by the SAE [27]. This adaption consists of six steps with increasingly higher requirements for the level of automation. Note that this is a sequential score, i.e. it is not possible to attain level 3 without having attained level 2, etc.:

1. No Automation (section A.2.2)
2. Assistance Automation (section A.2.2)
3. Partial Automation (section A.2.2)
4. Conditional Automation (section A.2.2)
5. High Automation (section A.2.2)
6. Full Automation (section A.2.2)

No Automation

Any drone which is fully controlled by the operator at any given time with only very basic functionalities is considered a non-autonomous drone. The operator of the drone has to be either physically present in the same area the drone is flying in, or a live video feed from the drone has to be transmitted to them. It is this level of automation that most consumer drones reside at.

Assistance Automation

Assisted drones can traverse simple environments controlled by the operator while providing a pose estimate for the operator. A complete description of the environment is available to the drone, and any actions the drone has to perform are executed by the operator. Similarly, it is up to the operator to decide which actions/tasks must be performed. Thus the most significant advancement, of the works at this autonomy level, is that the drone is capable of estimating its position.

Partial Automation

A partially automated drone maintains its ability to return pose estimates within the environment and gains the skill to map the environment while moving around. The

operator still has to manually move the drone to the inspection area but the drone can generate and follow a local inspection trajectory under the supervision of the operator.

Conditional Automation

At this automation level, the drone possesses the ability to observe and adapt trajectories as it traverses the environment. The interference of an operator is limited to specifying the type of environment and they only interrupt in rare situations. The main milestone is the capability to plan local actions such as trajectories to solve a task and the drone being capable of adapting to a changing and somewhat dynamic environment. In theory, it is, therefore, no longer required for the operator to maintain focus on a single drone, as the automation level is high enough for both traversing the environment, as well as adapting to changes in the environment.

High Automation

Employing highly automated drones, the role of the operator is shifted from using the drone as a tool, to managing the information gained from the task the drone is solving. Thus, the drone no longer requires monitoring and can initiate new tasks required by the operator. The drone itself will determine when, where, and what has to be done. At this level, the drone is capable of giving a consistent stream of information related to inspections of the vessel from which a continuous estimation of the condition can be made. It is up to the drone to ensure that the dynamic environment is explored and based on sensory information can solve the task it was given at launch. At this level, it is still required for the operator to initiate the drone to begin execution.

Full Automation

In the full automation category, the drone learns from past experiences to improve the execution efficiency of the task. The drone is able to perform under any conditions that an operator would normally operate under.

A.3 Assessment of Literature with regards to Expertise

The premise of being able to apply expert knowledge is the presence of information that can be interpreted. This information can come from a wide variety of sources, such as visual images, non-visual images, ultrasonic measurements, or even statistical methods used to describe areas at risk as a function of vessel age, type, previous assessment, etc. The process of applying expertise to this information can either be manual, semi-automatic, or automatic. Applying the expertise manually involves human surveyors assessing the sensory information and based on their experience, classifying the condition. Similarly, semi-automatic solutions can consist of tools that automate parts of the inspection process, such as easing the documentation process or by only fulfilling some of the defect metrics described in section A.3.

It is not only within marine vessel inspection that attempts have been made to automate the inspection process by capturing the expertise in the form of sensory input followed by an artificial assessment. Such examples can be found in railways [40], bridge structures

[41], wind turbine blades [42], tunnels [43]. Earlier surveys on inspection also date far back in time [44]. Thus, there exists a wide interest in automating the inspection processes, and since the elements used in many large construction efforts consist of roughly the same materials, e.g. steel and concrete, a solution not directly intended for marine vessel inspection may include the same components required for this application, too.

For this paper, however, we are focused only on existing literature work directly related to marine vessel inspection due to it being unfeasible to test all existing works on inspection processes on marine vessels. Like with many inspection processes, the inspection of marine vessels currently relies on visual inputs from the human surveyor, and thus many attempts at automating the inspection process uses visual cameras. The following section reviews the attempts to assess the condition of marine vessels by categorizing them into spatial-domain, wavelet, histogram, and deep learning methods. Finally, a subsection is reserved to present the works that attempt to evaluate the corrosion using non-visual methods, for the sake of completeness.

A.3.1 Spatial-domain based methods

Not only corrosion has been detected with image processing, but cracks and other types of deformations have received attention as well. In [31], a camera is used to capture information of a sample of metal, which is then analyzed using a combination of thresholding and morphology. The parameters for the morphology process are learned through a genetic algorithm. The main disadvantage of this method is the requirement to obtain a sample of the metal to be inspected. This makes it infeasible for large-scale vessel inspection.

Segmentation of corrosion in images is a task that desirably gives an exact location of the corrosion in the image. One of the ways to segment an image is to perform watershed transform [45]. One of the great challenges when performing a watershed transform for corrosion detection is segmentation, specifically when the image contains noise. An improved watershed transform is presented in [32] where the incorporation of the value and brightness of each pixel together with the canny operator was proposed [46]. The result was a more robust segmentation and less sensitivity to noise in the image. Canny edge detection has also been used directly for segmentation as shown in [47].

Saliency has been utilized in multiple occasions in [33, 48, 49, 50, 51]. The saliency map consists of a topographic map where bright values represent areas with defects and lower values represent areas with no defects. In [50], the saliency map is used as an input to two kinds of detectors: contrast-based and symmetry-based detectors. The two types of detectors are also combined to produce a single defect detector in three different ways using the logical operators **OR**, **AND**, and a custom or operator that averages the contribution of the contrast channels, intensity, color, and orientation as well as the symmetry map. One of the main results of this work is that their final defect detector is able to produce an Area Under the Curve (AUC, [52]) value of 0.8. Additionally, it was found that a contrast-based detector performs better than a symmetry-based detector, suggesting that contrast is capable of capturing more information used to discriminate between defective areas and non-defective areas.

An ensemble of different classifiers was explored in [53] ranging from a support vector

machine and Fisher Discriminant Analysis to K-nearest neighbor. The features used for the classifiers are all computed from the Hue, Saturation, and Value (HSV) color space. The main contribution is a combination of all the classifiers trained individually on the training features. Then, the final classifier, named PICARD, classifies inputs based on a majority vote of the in total 6 classifiers. Similarly, [54] used AdaBoost consisting of a linear combination of weak classifiers implemented using 48 Law's texture energy filters. 25 images were used to generate 39746 patches of which 12952 were labeled defective. 50% of the total amount of patches were used for training. One of the main results is a false positive and false negative of 17.16% and 3.39% respectively with the reasoning that it is more important to detect all defects than to have false positives. In Eich et al. [55] a global classifier is obtained by chaining two smaller classifiers together. The first classifier relies on the fact that corroded areas in images have a rough texture measured by the energy of the symmetric gray-level co-occurrence matrix. Based on the energy field, a threshold determines candidates for corrosion. The second classifier uses the output of the first classifier and the fact that corroded areas are bounded to the hue-saturation plane. Subsequently, a bidirectional histogram is built on which a filter is used to zero out entries that are 10% below the peak. After applying a Gaussian filter the remaining pixels are thresholded based on the filtered histogram.

A combination of traditional image processing techniques and neural networks has been explored in [37, 56]. Initially, they use a set of selected features in the image which is classified using a three-layer neural network with the goal of segmenting any corrosion defects in the image. The features are computed using a modified version of [57] and work by extracting the dominant colors of a small patch of the original image, which are fed into the neural network. A second experiment used k-means ([58]) initialized by K-means++ ([59]) to avoid clustering the k-means in the same area in the color space. Lastly, [60] presented an experiment where texture analysis is used to extract features and perform local binary patterns to weigh the difference in pixel intensities in the image. One of the main downsides of the different feature extractors used in [37, 56] is the required time to compute the features since they all operate on patches of the image, and the classification depends on the size and number of defects present in the image.

A.3.2 Wavelet based methods

Some of the early work on applying image processing tools on images of corrosion include [61, 62] in which a discrete wavelet transform is used. Initially, a three-level wavelet decomposition on an image is applied after which the image is divided into patches with a stride equal to the patch size. A feature vector of each patch is extracted consisting of the energy response of the wavelet transform at each frame. Finally, the classification of the feature vectors was performed by a 3-layer feed-forward neural network trained in a supervised setting. Though the application is corrosion on aircraft skin, it is included here for completeness since it uses the wavelet transform for corrosion detection, thus making it directly related to the application of marine vessel inspection considered in this paper.

Another example of the use of the wavelet transform was presented in [63], where corrosion located on the vessel hull was detected. The decomposition used in their work consists of applying all combinations of low- and high-pass filters and thereby obtaining four

sub-images from the one original image on which the process was repeated recursively. The image with two low-pass filters applied is also referred to as the image approximation, as it is just a smoothened version of the original image. The main contribution of that work is that the decomposition level is determined by computing the ratio between the Shannon entropy of the image approximation and the sum of the entropy of the other sub-images.

A.3.3 Histogram methods

Simpler approaches such as the ones in [64, 65] utilize a histogram of the image to determine the background based on the principle that the majority of the images will be without corrosion or other defects and, thus, they will be similar in color (due to the coating). The image is then thresholded based on the histogram to segment the defects.

Digital images are usually represented in the RGB color space, however, multiple other color representation models exist, some of which contain desirable properties for the classification of objects that are characterized by specific color [66]. In [67], images are interpreted in the Hue, Saturation, and Intensity (HSI) color space from which they build histograms of patches of the image with size 10x10 pixels. The histograms for each channel are treated as random variables on which they apply principal component analysis and varimax. The output of this process is features used for classification. Their work concludes that the mean H, median S, skew of S, and skew of I together with some physical characteristics such as the area, perimeter, length, and mean radius can be used to obtain a classification accuracy of approximately 85%.

A more comprehensive approach was used in [68] where external RGBD scanners are placed around the vessel when in drydock to create a 3D representation of the vessel including the color. Defects are then classified by considering a small area of the vessel surface and generating a histogram after the color channels have been converted to the HSV color space. Depending on how big the area that is corroded, segmentation happens based on either a histogram or a threshold.

A.3.4 Deep learning methods

Classification

Pure deep learning classification was also used in [69], however, corrosion is only detected on a binary whole-image level. So it can only be used to detect the presence of corrosion and not where in the image it is located. Though the training dataset size was very small, the method still managed to achieve an accuracy of 89.1%.

Object detection

More recent deep learning architectures have also been used for defect detection. Some preliminary results were presented in [12], where the authors have used transfer learning on a Single-Shot multi-box Detector (SSD, [70]) and a Faster R-CNN [71] with a VGG network [72] as a backbone. Instead of using the common intersection over union metric, an intersection of prediction was employed with the reasoning that detecting the presence of defects is more valuable than generating a bounding box that tightly fits the ground

truth. The findings indicate Faster R-CNN is better suited for corrosion detection than the single shot multi-box detector.

Deep learning architectures have also been used in [73] and the continuing work in [74, 75]. In the former, transfer learning was used to train a Faster R-CNN architecture with a VGG19 backbone and the system was able to detect three types of defects: surface-based corrosion and coating breakdown, edge-based corrosion and coating breakdown -, and non-coating-failure. In general, the work outlined in [73, 74, 75] presents a complete system including a drone and user interface for image processing and evaluation as well as the option for generating a report of the results. The main results of using Faster R-CNN with a VGG19 backbone is an accuracy of 81.37% when separating corrosion on welds and edges and 89.54% otherwise, which indicates deep learning approaches may be suitable for vessel classification. The same authors have tried to improve the accuracy of the system by introducing active human intervention in [34]. The human intervention is a human surveyor manually assessing an image that has been pre-processed by adjusting the brightness, gamma correction, and histogram equalization.

Not only general-purpose deep learning architectures have been explored for the detection of corrosion. In [76] a custom deep neural network was proposed to classify the level of corrosion on pipes as either None, Low, Medium, or High. To localize the corrosion within the image, the authors proposed to recursively sub-sample the input image if it contains medium or high levels of corrosion until the image size reaches a lower threshold. Consequently, this also means the number of inference calls on the classification network is high.

Segmentation

Some of the most popular deep neural networks have been investigated in [77] to identify the best suitable pipeline to perform inspections on robotic platforms with limited computational power. Specifically, two pipelines were proposed: The first one included a simple network performing binary classification on the image level and, if defects were detected, a larger network performing pixel-wise localization was deployed, upon the result of which, the image would be classified as *good, fair* or *poor*. Similarly, the second approach was to simply use a smaller network to categorize the image as *good, fair, poor* directly and only use a larger image network for localization. Based on their findings, the best performing pipeline is to do binary classification and perform overall category conditioning with a large pixel-wise segmentation. Additionally, among the investigated, architectures the Mask-RCNN performed the best, indicating that larger networks may be a necessity for high-quality inspection results.

One of the biggest challenges when employing deep learning approaches is the large number of annotated images required for creating a robust model. For this reason there exist multiple ways to annotate an image, some of which were explored in [78]. Specifically, a loss function based on the centroid loss was proposed which seeks to minimize the effect of weak annotations on the network architecture (Attention U-Net) [79]. They conclude that their approach to employing weakly supervised training produces similar results at a reduced annotation burden when compared to traditional full supervision in the task of detection of corrosion in images from marine vessels.

A.3.5 Non-visual methods

In this section, methods that do not directly rely on imagery are presented. The purpose is to include methods that use other sensory equipment than those naturally available to a human surveyor. The methods include using the grid method, ultrasonic, and augmented reality. The former is presented in [80] and is an approach to detect cracks by applying a grid pattern on the inspection surface, which in the specific experiments is a transparent sheet transferred to the inspection specimen. By analyzing the phase modulation of the light caused by the crack with Windowed Fast Fourier Transform, it is possible to separate the discontinuities in the phase with a simple threshold. Cracks as small as 5 μm in width are successfully detected and their location is determined with an accuracy of 1.2 mm. While the detection and localization accuracy are more than sufficient for the inspection of marine vessels, it has to be noted this method relies heavily on a fully controlled environment.

Other methods utilize ultrasonic measurements to measure the thickness of the steel. The advantage of this approach is an exact measurement of the integrity of the steel the vessel consists of. In [81], a robotic solution fitted with an ultrasonic sensor was presented. Though the sensor was mounted on a robotic platform, the only expertise in relation to the condition of the vessel was provided by a human surveyor.

The use of augmented reality has also been explored for inspection. Multiple applications for augmented reality were shown in [82]. Some of the applications highlighted by the authors are related to inspection by indicating faults and defects to the operator. The operator can also use augmented reality to point out faults and defects manually, and then save the location of the defect automatically. Augmented reality was also included in [83] where the utilization of a drone equipped with a live camera is presented. The operator is then shown the live camera feed.

More statistical approaches exist that rely on the physical process of corrosion such as the method proposed in [84]. The authors identify a set of parameters divided into operational and internal parameters. One of the operational parameters is the ballast ratio - the ratio between how long the vessel is loaded with cargo and the age of the vessel. Other parameters are trade route, coal corrosivity, and frequency of cargo changes. Thus, the operational parameters consist of the external forces applied to the vessel that influences its condition. On the contrary, the internal parameters describe the internal design measures taken to prevent corrosion. These include corrosion protection systems and structural member location and orientation. The conclusion of the work was a proposal to monitor the aforementioned parameters to build a corrosion rate database that would enhance the reliability of corrosion prediction models.

One of the most effective measures of preventing corrosion in marine vessels is by applying paint on the steel surface that prevents direct contact between the salt water and the steel. The disadvantage of coatings is the difficulty of being able to detect corrosion underneath them. Therefore, [85] proposed the use of microwave sensors to detect corrosion under paint and composite laminates. Their experimental setup included a painted steel plate on which they were able to detect corrosion even with varying paint thickness.

A.3.6 Summary

Table A.2 shows the main reported results for each evaluated paper. Note that these results are reported directly from the papers without scrutiny. Thus, a comparison is meaningless without referencing the underlying methods. It is included here for completeness as it gives an overview of what method the different papers have used to evaluate their results and how they have performed.

The scoring based on the weighting from the classification society and the scoring of the metrics for each work considered in this paper is listed in table A.3. It should be noted that while all of the works have some relation to the marine vessel inspection process, some of the presented works also include more general applications, such as crack detection on concrete structures and/or spalling.

From table A.3 it is clear that methods which rely on some form of deep neural network score better on the metrics used in this paper. Specifically, they can distinguish between more types of defects, alleviating the subjective assessment of the surveyor. Generally, all the methods investigated in this paper score poorly in quantification, i.e. they do not incorporate a way of quantifying the defects. Most of the methods use some form of in-image localization. This approach has the advantage of only requiring a single image, however, it limits the quantification to be strictly on the image plane. Since the area affected by a defect will be perceived drastically differently depending on the perspective and viewing angle, it is difficult to get a real-world quantification of the defects. One of the advantages of using camera images is the high level of automation that can be achieved as well as the flexibility. Cameras can be small in size so as to fit in even tighter places while only requiring a light source when there is no natural light present, such as when inspecting a cargo hold or a vessel hull.

Another conclusion from the table is that the differentiation between defect types has been neglected. A common trend has been to find corrosion or cracks, with none of the investigated methods addressing the need for also detecting deformations in the steel of the vessel. These types of damage are not infrequent and can occur when the anchor hits the hull of the vessel or when loading/unloading cargo. Some of the more severe deformations can occur when a vessel is sailing in low depths or due to collisions - the latter accounting for 35% of marine vessel accidents [90].

The attention the marine vessel inspection process has received over the last 35 years — from an automation perspective — has been increasing steadily as indicated by figure A.5. The number of publications shows an increasing interest in the topic, and with the recent advances in deep learning and computer vision, the level of expertise that can be reproduced autonomously is steadily rising according to the metrics defined in this paper.

To achieve a higher expertise score, it is clear there has to be a bigger focus on quantifying any detected defect. Detecting different kinds of defects has been addressed in some cases with the use of deep learning that is capable of distinguishing multiple types of defects, and in some cases providing an in-image segmentation of the defect. While this provides some form of location, it is not directly possible to transform a single image coordinate into a usable 3D coordinate that can be used by repairmen to localize the defect. As technology advances, the range of sensors capable of capturing 3D information increases.

	Evaluation method	Results
Siegel et al. [61]	Visual evaluation	Demonstration
Qaddoumi et al. [85]	Visual evaluation	Demonstration
Siegel and Gunatilake [62]	Visual evaluation	Demonstration
Zheng et al. [31]	Hole and crack detection accuracy on dataset	holes 91% cracks 86%
Gardiner and Melchers [84]	Parameters affecting corrosion	Degradation hierarchy
Avril et al. [80]	Metric size of cracks	smallest detection 5 µm
Choi and Kim [67]	Accuracy on dataset	corrosion 85%
Akinfiev et al. [81]	Not stated	Not stated
Zaidan et al. [47]	Visual evaluation	Demonstration
Navarro et al. [64]	Classification rate	92.5%
Bonnin-Pascual and Ortiz [54]	Confusion matrix	FP=20.47 rate, FN=20.91 rate, FP=17.16%, FN=3.39%
Ji et al. [32]	Not stated	Not stated
Jahanshahi et al. [86]	Metric size of cracks	Maximum deviance from ground truth <15mm
Navarro et al. [65]	Accuracy on dataset	95%
Fernández-Isla et al. [63]	Confusion matrix	FP=6.8 error, FN=0.9 error
Eich et al. [55]	Confusion matrix	Corrosion: FP=9.8%, FN=5.86% Crack: FP=0.72%, FN=0.52%
Bonnin-Pascual and Ortiz [33]	AUC	0.9
Dini and Mura [82]	Not stated (augmented reality)	Not stated
Papachristos and Alexis [83]	Not stated (teleoperation)	Not stated
Bonnin-Pascual and Ortiz [49]	AUC	0.8
Bonnin-Pascual and Ortiz [48]	AUC	0.9
Ozog and Eustice [35]	Not stated	Not stated
Petricca et al. [69]	Confusion matrix	Accuracy=92%
Aijazi et al. [68]	F1 measure	0.90
Ortiz et al. [56]	Success rate	0.87
Ortiz et al. [37]	F1 measure	0.92
Ortiz et al. [87]	Visual evaluation	Demonstration
Yang et al. [88]	F1 measure	Accuracy=97.96 Precision=81.73 Recall=78.97 F1=79.95
Maglietta et al. [53]	Accuracy on dataset	Corrosion 0.961
Ortiz et al. [12]	Intersection over Precision	Visually inspected on graph
Bonnin-Pascual and Ortiz [50]	AUC	Visually inspected on graph
Liu et al. [73]	Confusion matrix	Accuracy=81.37% on 5 classes
Liu et al. [74]	Confusion matrix	Accuracy=89.54% on 5 classes
Liu et al. [75]	Confusion matrix	Accuracy=89.54% on 5 classes
Bastian et al. [76]	Accuracy and F1	Accuracy=98.2, F1=96.73
Liu et al. [34]	Confusion matrix	Accuracy=89.54% on 5 classes
Hoskere et al. [89]	Accuracy on dataset	91.7%
Andersen et al. [77]	Intersection over Union	0.156
Yao et al. [78]	Intersection over Union	0.7542

Table A.2. Reported results for each evaluated paper. Note that the results reported here are taken directly from the respective papers. Thus a direct comparison between the results is of little value without also considering the underlying methods and metrics.

	Method	Type	Location	Quantification	Autonomous evaluation	Flexibility	Expertise score
Andersen et al. [77]	DL	1.0	0.9	0.3	0.5	0.9	0.814
Liu et al. [75]	DL	0.9	0.8	0.3	1.0	0.9	0.800
Hoskere et al. [89]		0.9	0.8	0.3	0.9	0.9	0.793
Liu et al. [74]	DL	0.8	0.8	0.3	1.0	0.9	0.771
Liu et al. [34]	DL	0.8	0.8	0.3	1.0	0.9	0.771
Jahanshahi et al. [86]		0.5	0.8	1.0	0.5	0.9	0.750
Bonnin-Pascual and Ortiz [33]	Spatial	0.8	0.5	0.1	0.9	0.9	0.671
Ortiz et al. [87]		0.8	0.5	0.1	0.9	0.9	0.671
Liu et al. [73]	DL	0.9	0.5	0.1	0.5	0.9	0.671
Yang et al. [88]		0.5	0.5	0.5	0.9	0.9	0.643
Yao et al. [78]		0.6	0.5	0.3	1.0	0.7	0.593
Fernández-Isla et al. [63]	Wavelet	0.5	0.5	0.1	0.9	0.9	0.586
Bonnin-Pascual and Ortiz [49]	Spatial	0.6	0.3	0.1	1.0	0.9	0.579
Bonnin-Pascual and Ortiz [48]	Spatial	0.6	0.3	0.1	1.0	0.9	0.579
Bonnin-Pascual and Ortiz [50]	Spatial	0.6	0.3	0.1	1.0	0.9	0.579
Bonnin-Pascual and Ortiz [54]	Spatial	0.5	0.5	0.0	0.9	0.9	0.571
Eich et al. [55]		0.6	0.3	0.1	0.9	0.9	0.571
Ozog and Eustice [35]		0.1	1.0	0.8	1.0	0.5	0.571
Bastian et al. [76]		0.5	0.7	0.6	0.5	0.5	0.557
Ortiz et al. [56]	Spatial	0.5	0.5	0.1	0.5	0.9	0.557
Ortiz et al. [37]	Spatial	0.5	0.5	0.1	0.5	0.9	0.557
Akinfiev et al. [81]		0.5	0.5	1.0	0.8	0.3	0.536
Ortiz et al. [12]	DL	0.5	0.3	0.1	0.5	0.9	0.514
Choi and Kim [67]	Histogram	0.5	0.1	0.1	0.9	0.9	0.500
Aijazi et al. [68]	Histogram	0.5	0.9	0.9	0.5	0.0	0.500
Navarro et al. [64]	Histogram	0.5	0.5	0.3	0.8	0.4	0.464
Navarro et al. [65]	Histogram	0.5	0.5	0.3	0.8	0.4	0.464
Petricca et al. [69]	DL	0.5	0.0	0.0	0.5	0.9	0.436
Avril et al. [80]	Non-visual	0.5	0.5	1.0	0.1	0.0	0.400
Ji et al. [32]	Spatial	0.4	0.5	0.8	0.5	0.1	0.400
Zaidan et al. [47]	Spatial	0.5	0.3	0.0	0.5	0.5	0.386
Siegel et al. [61]	Wavelet	0.5	0.8	0.1	0.1	0.1	0.364
Siegel and Gunatilake [62]	Wavelet	0.6	0.5	0.1	0.5	0.1	0.357
Papachristos and Alexis [83]	Non-visual	0.0	0.8	0.1	0.1	0.5	0.336
Dini and Mura [82]	Non-visual	0.0	0.8	0.1	0.0	0.5	0.329
Maglietta et al. [53]	Spatial	0.5	0.5	0.3	0.5	0.0	0.329
Zheng et al. [31]	Spatial	0.5	0.5	0.0	0.1	0.0	0.257
Qaddoumi et al. [85]	Non-visual	0.5	0.3	0.1	0.0	0.0	0.221
Gardiner and Melchers [84]	Non-visual	0.5	0.0	0.5	0.0	0.0	0.214

Table A.3. The score of all references. The metrics are used to compute the final expertise score according to equation A.1. The expertise scoring lies in the interval $[0-1]$ with a higher value meaning a better fit between the expectations of the classification societies and the actual level of expertise captured.

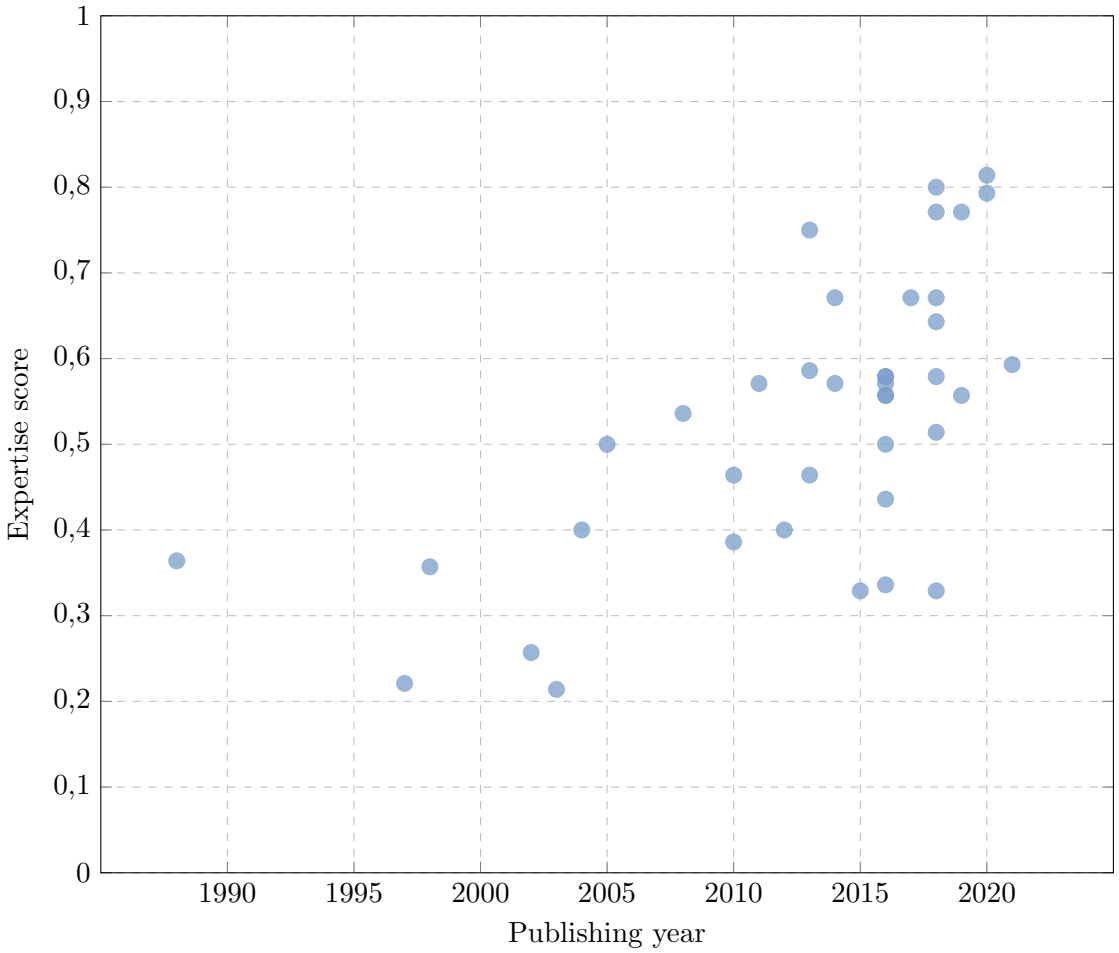


Figure A.5. The attention the marine vessel inspection process has received over the last 35 years from an automation perspective has been increasing. This is an indication of the technological advancement that has been made within image processing and computer vision, as that is by far the most popular approach to automating marine vessel inspections.

A simple solution to the localization problem would simply be to use a stereo camera setup that can provide a simple form of 3D localization [7]. The disadvantage of stereo vision is possibly the difficulty in achieving high enough accuracy to also perform quantification. Quantifying the amount of corrosion can be done on a higher level as was shown in [91, 92] however, both the severity of deformations and cracks have to be quantified using physical distance units. This might require a higher level of detail than a stereo vision camera setup can provide. Unless an orthophoto can be produced, a better approach might be to use a laser range sensor that is capable of producing high-accuracy point clouds.

A.4 Evaluation with regards to the Engineering Aspect

Autonomous robotic platforms for vessel inspection are still in their early stages, and few systems provide a high level of mobile autonomy. This section will cover the level of autonomy that is currently available using a taxonomy inspired by the SAE taxonomy [27] (see figure A.6). The SAE taxonomy was originally intended for motor vehicles on the roadway system, however, the same reasoning can be applied to any autonomous vehicle. Thus, the gradual removal of the operator is similar for both grounded vehicles and aerial

vehicles, ultimately progressing to full autonomy without the influence of an operator [93]. In the context of drone applications in confined spaces, the specific adaptation used here is described in section A.2.2. For each level of autonomy, the relevant literature will be reviewed and categorized by the relevant taxonomy of automation. The levels do not implicitly describe the robustness of each system. Two systems in the same category can perform differently with regards to — for example — accuracy, precision, and how they handle disturbances. Likewise, the levels will not differentiate whether the drone is airborne, sub-sea, or ground-based.

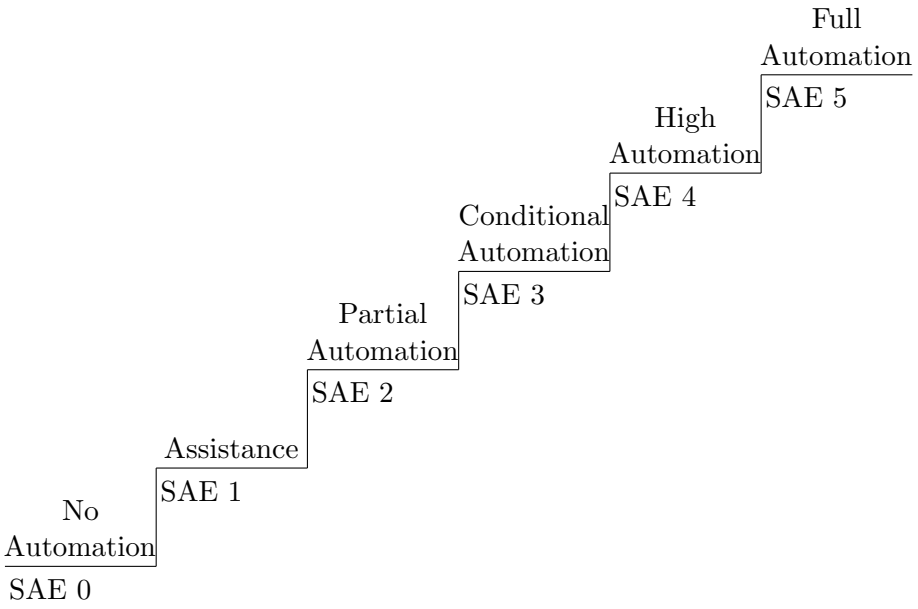


Figure A.6. The autonomy taxonomy used here is sequential, starting at no automation and progressively decreasing the involvement of the operator to the point of a self-sustainable solution with full automation.

Drones/Robotic platforms which are not aimed at vessel inspections, but can provide a comparably high level of autonomy, will also be reviewed with the purpose of a future outlook on what research could be applied to conduct autonomous vessel inspections.

A.4.1 SAE 0 - No Automation

The first category, where the human is in control of all parts of the navigation during the vessel inspection, primarily consists of legged and so-called crawler platforms operating above the sea surface. Small 4-legged robots are presented in [94], where magnets are attached to each foot to ensure the capability of climbing vertical surfaces. Furthermore, the authors enhanced its capability to climb on industrial beams with a thickness of less than 5 cm, and through narrow gaps of 23 cm. Other approaches use magnetic wheels [95, 42, 55] or chain-like tracks with inserted magnets in each link [96]. The authors in [97, 98], have investigated the possibility of a rubber-wheeled robot with top-mounted propellers to provide a thrust that increases the friction between the wheels and the surface. The hybrid actuation approach adds the advantage of climbing over vertical walls made of non-magnetic materials, such as glass fiber, wood, and aluminum. A manually controlled Unmanned Aerial Vehicle (UAV) was used in [99] to collect image data of a wall from a cargo hull of a container vessel. These images were then offloaded from the drone and used

to build a mosaic of the wall, with the benefit of an increased overview of the inspection area.

None of the aforementioned research efforts in this category has capabilities to aid in autonomously inspecting the vessel.

A.4.2 SAE 1 - Assistance

Within this level of autonomy, the operator enjoys some level of assistance in controlling the robotic platform, which could be in the form of localization and/or a stabilization system. Examples of localization systems are the relative pose estimations for sub-sea hull inspection described in [100] and [101]. The authors of [101] and [102] use stereo vision to get a relative pose by matching image feature points between the camera views and frames. Both systems are tested on a real vessel, where the authors in [101] tested their system on a 19x2 meter hull section of a real marine vessel. The system was verified by showcasing the reconstruction of the hull section using the pose and images along the inspection path. Another approach by [35] uses the CAD model of the hull in combination with a camera and a Doppler velocity log (DVL), to localize itself with respect to the coordinate frame of the hull. By, furthermore, adopting a bundle adjustment system the authors built a mosaic map overlayed on the CAD model of the hull.

Absolute localization systems for UAVs operating in the confined spaces of the vessel were recently investigated in [2, 3, 4]. Employing a combination of Visual Inertial Odometry (VIO) and detection of existing known structural 3D landmarks were used in [2] to estimate the absolute pose in a mockup model of a water ballast tank. This had the advantage of only requiring a high-level map of the environment. Deep neural networks were used in [3, 4] to generate accurate 3D feature descriptors within an existing point cloud map and for the current viewpoint of the vehicle. The feature descriptors were then matched and used together with a visual-inertial odometry estimate in an extended Kalman filter, resulting in an absolute pose estimate.

A magnetic track/belt robot was proposed by [103], which uses computer vision to localize the robot and create a mosaic map of the traveled trajectory. The system was tested in the cargo hulls of a bulk carrier, and was according to the experiments, able to combine images and create a 2d overview map of approximately 1 x 3 meters of a section of the wall of the cargo hull. Similarly, [104] developed a magnetic track/belt robot capable of performing thickness measurements on the dry parts of a hull on a marine vessel. The robot was able to maintain an estimation of its position using an extended Kalman filter to fuse wheel encoder readings and IMU data.

A.4.3 SAE 2 - Partial Automation

Most subsea inspection systems are naturally focused on the outer hull of the vessel. The researchers in [105, 106, 107] focus on hull mapping with simple path planning systems built into the commercially available ROV. This research involves pose graph optimization based on pose estimation by fusing multiple sensors, namely cameras, sonars, and DVLs. The executed trajectory near the hull structure is a simple zig-zag pattern, also known as meander pattern. A noticeable addition is the ability to align previous and years-old

inspection scans with recent scans [106]. This alignment adds important value to the inspection data since the rate and progress of any deterioration or biofouling of the hull can now be monitored, which in turn can be used to estimate required inspection intervals. Specific to this category is also the focus of added capability of path planning or execution, where [36, 108] presents a method for full coverage inspection of a hull through redundant roadmaps [109]. They also specifically address the challenging areas around the driveshaft and propeller at the stern of the vessel and illustrate the possibility for full hull inspection with a resolution in the order of 10cm [36]. The novelty within the planning aspect specifically lies in the redundant roadmaps where previous subsea inspection methods have applied a mostly uniform zig-zag pattern. Above sea level, a magnetic wheeled robot, Sparrow, has been developed in [110] to conduct contact-based ultrasonic thickness inspection. The localization system was based on Marvel Mind beacons, attached to the hull of the vessel, and required line of sight to the robot to provide stable position estimates. The Sparrow can autonomously move in simple zig-zag patterns during the inspection, using position estimates from wheel odometry and the beacons. This capability was tested within a 0.7x0.7m area. Airborne solutions have in recent years been researched as viable solutions for the visual inspection of marine vessels. Examples of this are the AscTec Firefly, the Hummingbird and the Pelican platforms which were used in [37, 48, 87]. To navigate, the system presented in [37] utilizes LIDARs, cameras, and IMU data from the onboard AscTec flight controller. It makes use of a 2D Hokuyo UST-20LX lidar to estimate distances to the surrounding walls and a 1D LIDAR-Lite to estimate the height above the floor. Experiments were carried out in the cargo hull and the top ballast tanks of a cargo vessel. 2D scan matching was used to align the UAV with the surrounding walls and, in combination with the LIDAR-lite for height estimate, the system was able to maintain the position of the UAV inside the confined space. The autonomous capabilities of the works in [48] and [87] are similar to [37], due to the employment of the same positioning system to maintain the robot's position next to the wall of the cargo or ballast tank. In [87] the authors added a multi-threaded Binary descriptor-based Image MOSaicing (BIMOS) approach to create an overview of the inspection surface, using ORB features and Keyframes.

A.4.4 SAE 3 - Conditional Automation

The aerial system used in [111], includes self-localization, navigation, and obstacle avoidance. The authors, furthermore, add a Safety Manager system which is built on top of the obstacle avoidance capabilities, by (1) preventing the drone from flying too close to the ceiling, and (2), automatically landing the UAV when the battery voltage gets below a threshold or (3), hover the drone when the wireless connection to its base station is lost. The system is designed for confined spaces within marine vessels, but the experiments are conducted in an office-like environment of approximately 9x8x2.5 meters. Obstacles were simulated using cardboard pillars, and the UAV was successful in planning paths, in 2D, around the obstacles. The system was later updated in [112], to make the platform as usable as possible for non-experts. The level of autonomy is the same as in [111], but the focus was on the human interaction with the UAV using the Supervised Autonomy approach from [113]. The work in [111] adds a human control interface to the UAV, where the operator can overrule part of the automation layers, i.e. path execution. During the human

operation, the UAV will, however, overrule the operator, and perform evasive actions, in case obstacles are in the human executed path. The authors tested the system in an office-like environment, showing the obstacle avoidance capabilities, where the system prevents the operator from flying into a wall. The authors of [114] propose an aerial autonomous system for fire detection in visually degraded confined spaces of marine vessels. In terms of localization, the authors employ an absolute pose estimation method that fuses Visual Inertial Odometry estimates with 6-DOF pose estimates, based on point cloud data from an RGB-D camera with a known map of the environment. Online motion planning is applied, which combines A* path planning with a receding horizon control framework for obstacle avoidance. The majority of their tests were conducted in a corridor onboard a vessel, with outbreaks of fires along a corridor. The fires are automatically detected using a FLIR infrared camera using a threshold for temperatures above 100 degrees C. The UAV was successfully able to automatically navigate along the corridor and through the doorways of the test environment.

A.4.5 SAE 4 - High Automation

The state of the current research has to our knowledge not yet reached **[SAE level 4]** autonomy within marine vessels. However research within similar environments could reasonably well be applied to raise the autonomy level. The research mentioned in this section is, therefore, most of a higher autonomy level than in the area of marine vessels.

The authors in [115] presented a path planning framework for autonomously underground mine exploration, based on random trees while also accounting for the robots' endurance limit. Their approach is field tested on a UAV in an underground mine. The authors, furthermore, improve the system and presented a solution in [116] for underground mine rescue using the same UAV. Their system is capable of autonomously exploring the unknown mine environment and detecting and localizing objects of interest, which in their case were humans in need of rescue. Their approach is further expanded in [117] to include other vehicles, more specifically the legged robot ANYmal [118] from ANYbotics AG in Switzerland. ANYmal was also used to perform autonomous inspection inside offshore platforms in [119]. Here, it was shown that ANYmal could perform some of the same simple inspection tasks as the human surveyors, such as using its manipulator to push buttons, toggle switches and fuses, and turn valves. However, it still lacked behind the humans in simple tasks such as opening and closing doors, which was required due to fire prevention onboard the platform.

A.4.6 SAE 5 - Full Automation

Currently, there is no fully autonomous system, category **[SAE level 5]** which is relevant for vessel inspections.

A.4.7 Other important aspects

Little research effort within automating inspections has focused on addressing the requirements for surface preparation before an inspection can be performed. For visual and ultrasonic testing this is most often a thorough cleaning of the surface, which is currently

most often done by humans. The authors of [120] designed an inspection and cleaning system for removing biofouling on the hull of the vessel.

A.4.8 Summary

Though robotics platforms have been used in the inspection process of marine vessels, many of the platforms do not achieve a high level of autonomy and still rely on human operators for a significant part of the operation. Many of the platforms presented are based on general concepts such as a flying drone or a submarine. This independence also means there is a high probability of being able to transfer a platform from other applications where a higher level of autonomy has been achieved to the marine vessel inspection process. Examples of this were shown to achieve an adopted SAE level of 4 - higher than any of the investigated works that specifically target marine vessels.

A.5 Overall assessment of the state of the art

A visual interpretation of the existing work is shown in figure A.7. From this figure, it is visualized that there is a lack of systems that encapsulates both the engineering and expertise aspect that is necessary to alleviate the human involvement in the inspection process.

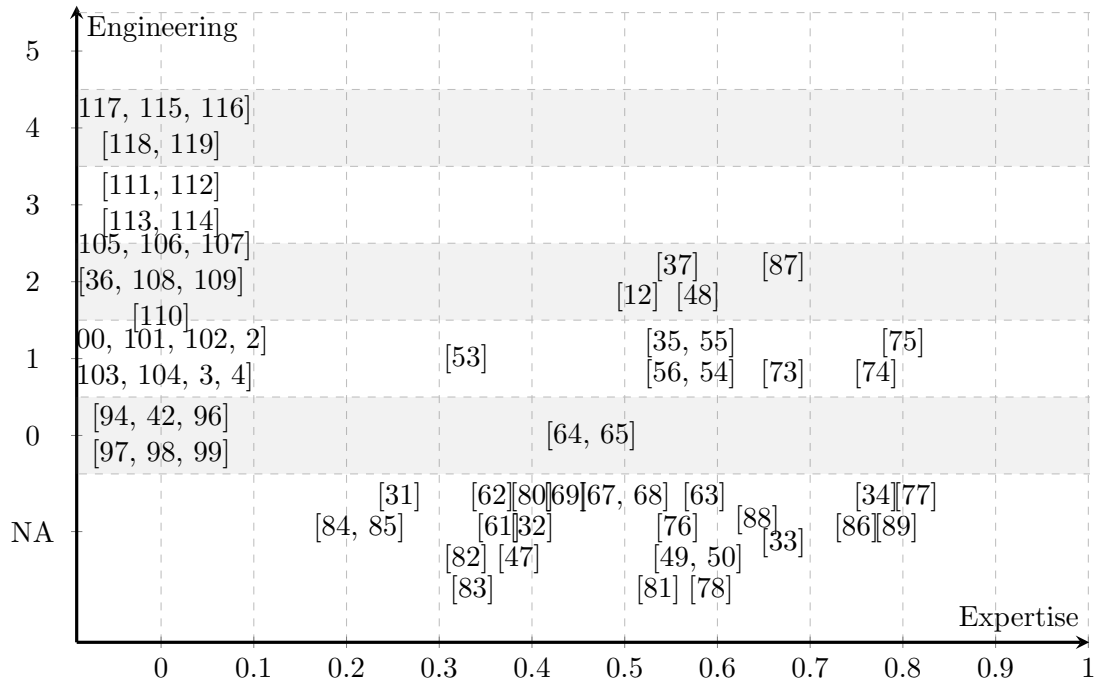


Figure A.7. The existing works scored on their expertise and engineering aspect. The category NA captures all the works that presented some form of expertise, but have not presented a way of collecting the sensory information necessary for a fully autonomous system.

It should be noted that some of the references in figure A.7 are not directly trying to solve the challenge of autonomous marine vessel inspection, however, they all show some form of potential.

While most of the literature included in this paper tries to solve either the expertise or engineering aspect, some of the works are part of a larger project that collectively tries to solve the challenge of automating the inspection process. Generally, it is these papers consisting of collections of previous work that score high in both expertise and engineering.

Among the highest scoring works are a significant number using UAVs as a robotic platform to reach otherwise inaccessible areas. One reason for this might be the ease of access to these kinds of platforms in combination with the low requirement for specific external environments — i.e. they don't rely on the presence of water and work in non-metal structures. Though they are limited by their heavy trade-off between lifting capacity and battery life, they can quickly traverse large areas while carrying lightweight sensor equipment such as cameras. In return, the cameras can provide a large amount of information about the environment where the drone is located and can be used to detect defects. By using stereo vision, the drone can even achieve 3D vision that enables localization of defects as was shown in Brogaard et al. [7]. A general challenge when relying on cameras as the input for detection is the reliability of external lighting. In some areas of a marine vessel, such as in the ballast tanks, there is no natural light, thus the drone has to carry it on-board further reducing the effective flight time, in the case of aerial drones.

Some of the works presented in this paper utilize multiple robotic platforms to perform different parts of the inspection process. This means the platforms can be more specialized in solving one part of the inspection process. It does also mean that the output of the technologies has to be merged before a complete overview of the vessel condition can be given, thus demanding a higher level of cooperation.

Many of the works reviewed in this paper address both capturing sensory information through the use of cameras and drones while also presenting some level of image processing to evaluate the sensory information. Only a few of them address the challenge of capturing image data in areas where there are very poor lighting conditions. Since many areas of modern marine vessels have little to no natural light, any drone has to carry any necessary light sources with it which complicates the detection. Additionally, none of the methods address the uncertainty associated with detection. I.e. the detection's not influencing the behavior of the drone to optimize the ability to accurately do defect quantification – a critical step towards an autonomous solution that provides the necessary level of quantification for the classification society. Similarly, there is a clear trend to use object detectors and segmentation models to provide a deterministic output. By doing so, they fail to encapsulate the ambiguity that is inherently present in visual defect detection where information is only partially observable due to the surface conditions where the defects are located. Instead of these deterministic approaches, one future direction for marine vessel inspection that has to be addressed is a distribution estimation over the detected faults and defects. If a classification society has to rely on the detections without being present, they need access to some form of certainty metric from which they can assess the overall vessel condition without risking a wrongly tuned confidence threshold altering the vessel classification.

Though a wide range of detection models has been tried on defect detection, very little emphasis is put on processing the detections in a human-readable manner - e.g. by converting the mapped detections to the reference frame of the vessel by noting on which

longitudinal or web frame the defect is present on. This kind of topological representation could also be used for navigational purposes. The vessels are well structured and the layout of all areas to be inspected is usually known in advance. This information is rarely used in combination with active defect detection to optimize the information gathered during the traversal of the inspection areas. Using a topological navigation planner would also alleviate the risk of drifts. If the drone is relying on a high-resolution map representation of the environment, drifting increases the risk of collision with the environment. By using a topological planner that relies only on high-level information such as longitudinal counts, web frame counts, stringer level, etc. in combination with local information like VIO and 3D sensory information, the need for high-resolution maps can be reduced. An instance where relying on internal map representations can be insufficient is when navigating to an unexplored area through a narrow passage. If the map representation is too low, the narrow passage is not visible, or difficult to reliably sample paths during exploration. Simultaneously, a too high map resolution increases the requirement for onboard processing.

A.6 Conclusion

In this paper, works on autonomous inspection were reviewed and evaluated with respect to a series of metrics defined to indicate how far the state of the art has progressed when compared to the needs of the classification societies. The metrics consist of two parts: Expertise and Engineering. The purpose of this split was to separate the part of an autonomous system that performs condition assessment of vessels and the part that enables the inspection to be physically carried out.

The metrics defined for the expertise were intended to show how well a given method can assess the types of defects that are typically found during a marine vessel inspection as well as quantify them. Generally, most of the methods struggle to quantify the defects they are able to detect. While the detection of defects is useful as a starting point, it is not sufficient as a replacement for manual inspection.

The adapted SAE taxonomy was used as a metric to evaluate the robotic platforms related to marine vessel inspection. This step-wise taxonomy describes the level of autonomy of each robotic platform where a higher score means a lower level of manual operation involved in the inspection. Some of the robotic platforms achieved a significant level of autonomy.

Though the purpose of this paper was not to in-depth review the machine learning and locomotion methods used in the literature, we have observed a high correlation between using deep learning techniques in a system and scoring high in our metrics – specifically the expertise one. In combination with flexible robotic platforms, such as UAVs, deep learning has proven to be an effective approach to the marine vessel inspection process.

Based on the analysis done in this paper we have identified some crucial future research directions within marine vessel inspection and classification: a higher level of quantification, human interpretable defect localization, defect-aware navigation, and the notion of probabilistic inspection. Remote inspection cannot be realistically implemented as an assistive tool for human surveyors unless a higher level of quantification is achieved. By doing so, the expertise level of the solutions would be increased significantly. Additionally,

defect localization in a human interpretable manner would be directly usable in the vessel condition documentation. This would help increase the confidence of the surveyor and would reduce the amount of translation that has to be done between users of the new autonomous system and the established classification process. As a further note, there is a lack of navigation methods that incorporate faults and defect detections to increase detection confidence and quantification accuracy, for documentation purposes, or otherwise better coverage of an area under consideration. From the analysis, we also found a lack of works that address the ambiguity of the classification process. Future research should help address this ambiguity by relying less on single binary ground truths as these are both difficult to access and rely on due to residue build-up in the ballast tanks.

We believe that these four areas of research would significantly increase not only the autonomy of marine vessel inspection but also increase the confidence of the human surveyors that have to rely on the system and ultimately are responsible for the vessel certification.

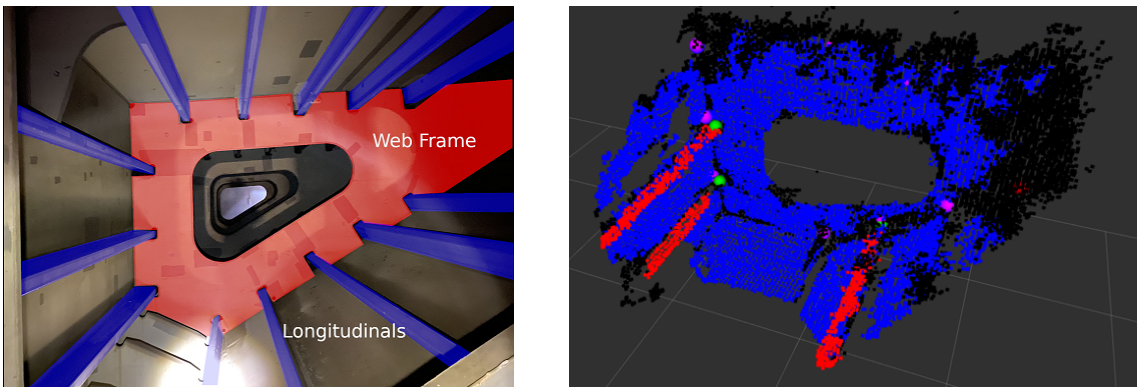
A.7 Acknowledgements

The work has been conducted under the Inspectrone project funded by Innovation Fund Denmark (IFD). The authors of this paper would like to thank Lloyd's Register for sharing their expertise in corrosion classification and maritime inspection standards. Additionally, we are thankful to Dampskibsselskabet NORDEN A/S for providing first-hand experience in the classification process on their vessels.

Chapter 3

Absolute Localization with custom structural landmarks

Confined industrial spaces are often dark and visually feature poor, resulting in sub-optimal performance of Visual-Inertial Odometry (VIO). When considering the ballast tanks of marine vessels and many other industrial spaces, a lot of repetitive structure and symmetry is however often present. For ballast tanks, a range of compartments of similar dimensions are coupled in a grid-shaped structure along the bottom and side of the vessel, also shown earlier in Figure 1.2. The internal structure of each compartment tends to be well defined and, therefore, an initial effort is made to investigate the use of prominent 3D structural elements as known landmarks in confined spaces. The next article presents such a method in which custom structural landmarks are detected and extracted from a water ballast tank. The structural landmarks are then registered with the same landmarks in a high-level map to provide an absolute pose estimate. With this approach, there is no need for detailed maps of the environment, but only high-level information, in the form of landmarks, from the structural drawings of the ship. This can simplify the required landmark points and the need for detailed maps. For our case with the ballast tanks, the intersection between the web frame and the longitudinal was used, where Figure 3.1a highlights these structural elements as the red and blue areas, respectively. An example of a point cloud generated



(a) *(b)*

Figure 3.1. Illustration of the longitudinals and web-frame inside a water ballast tank (a) and the longitudinals highlighted in a pointcloud scan from an RGB-D sensor (b). The purple and green dots indicate the intersection points between the longitudnials and web-frame in the map and the scan respectively.

by the depth camera of the UAV is shown in Figure 3.1b, where the green landmarks indicate the detected intersection points and the purple dots the position of landmarks manually extracted from the drawings of the vessel. Using the 2D drawings of the vessel to generate a simplistic map of the environment can limit the detail requirements of the map and simplify the registration of the map to the environment. Since each compartment of the ballast tanks consists of the same high-level structural elements, a list of high-level landmarks could be kept onboard the UAV in a hierarchical setup. In **article B**, we demonstrate a proof-of-concept toward such a custom structural landmark approach for localization in confined spaces. The primary contribution of the article was the creation of a custom feature extractor with the addition of a simple hierarchy setup to locate defects in water ballast tanks. The core of the system consists of the Visual Inertial Odometry system, ROVIO. The absolute pose estimate from the aforementioned landmark estimation system is then used to provide absolute corrections to the VIO pose estimate. An overview of the entire pipeline can be seen in Figure 3.2, where the desired output is the robot pose estimate and the defect pose estimate.

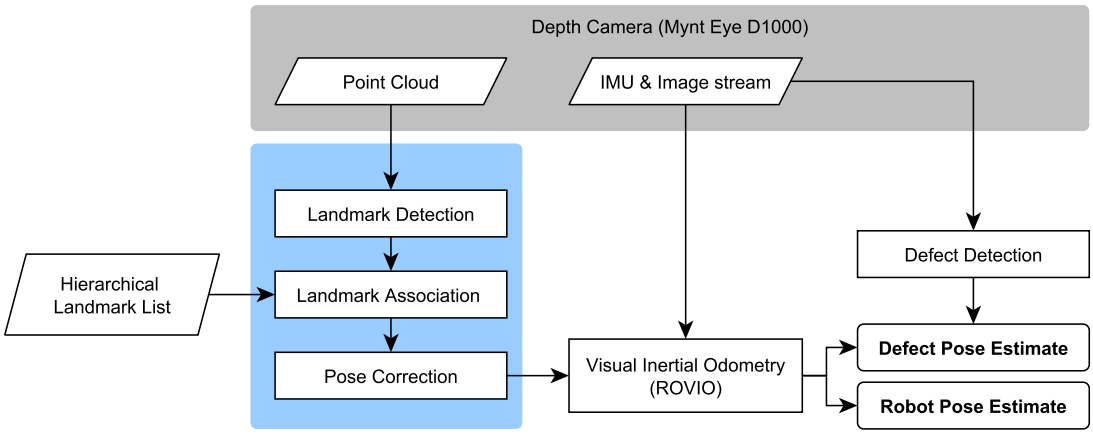


Figure 3.2. The absolute localization and defect pose estimation pipeline using custom landmark extraction from the environment and visual inertial odometry.

The proposed method was tested using a small UAV with an onboard stereo camera, which provided both the point cloud for the landmark extraction and the image and Inertial Measurement Unit (IMU) stream for the ROVIO based VIO estimate. The results of the article showed an improvement in pose accuracy compared to using the raw VIO estimate. Furthermore, the system was used to localize simulated defects installed within the mock-up model described in chapter A.1. Additionally, the results showed that the system was able to accurately determine the absolute pose of the defects using the proposed method. This approach can be useful in scenarios where only simple 2D drawings are available compared to newer CAD files. In addition, it adds well-defined landmarks that could be tuned to perform well for each given environment. The method described in **article F** uses a similar approach to extract a structural element from a chemical distillation column to determine whether the robot is in a reliable position for its localization estimation. However, the custom landmark extraction also makes it somewhat less generalizable to other structures without human-created feature extraction methods. The approach in **article B** also requires a well-structured extraction of 3D intersection points, and could face issues with asset scalability due to a potential time-intensive manual extraction process

from the structural drawings. Despite the listed potential disadvantages, the results from the article, show that the system was able to accurately estimate the position of the UAV and the detected defects in the water ballast tank.

Article B

Towards UAV-based absolute hierarchical localization in confined space

Rune Y. Brogaard[1], Marcin Zajaczkowski[1], Luka Kovac[2], Ole Ravn[2], Evangelos Boukas[2]

[1]FORCE Technology, Park Allé 345, Brøndby, Denmark

[2]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

Abstract:

To perform safe, autonomous, visual inspections of maritime vessels, accurate and lightweight localization of UAVs in dark and confined areas is required. Such visual inspections are performed by expert surveyors, periodically, and aim at the classification of the vessel by identifying defect locations within the frames of the ship. The paper at hand presents our initial efforts in absolute localization as a solution to the aforementioned problems using a UAV. Employing a combination of Visual Inertial Odmetry and detection of existing known structural 3D landmarks, an absolute and hierarchical localization system is created. With this approach there is no need for detailed maps of the environment, but only high-level information —i.e. few landmark points— from the structural drawings of the ship. The system is tested within a mock-up model of a topside water ballast tank, and its localization is compared to a groundtruth, which is estimated based on ArUco markers mounted on the walls inside the model. Our initial results indicate that the use of geometric landmarks, extracted from vessel designs, greatly benefit the UAV localization accuracy as well as the localization of defects with respect to the vessel body frame.

B.1 Introduction

The inspection of ballast tanks is an important factor in assessing the integrity of the ship and, therefore, the safety of the crew and the environment during operation. These inspections are mostly carried out by humans, which involves entering into confined spaces. Inspection is considered dangerous work, due to the variability's in the atmospheric conditions. Likewise, this line of work is also often tiresome, dirty, and dangerous, which increases the risk of human errors, such as missed critical defects. It is important to

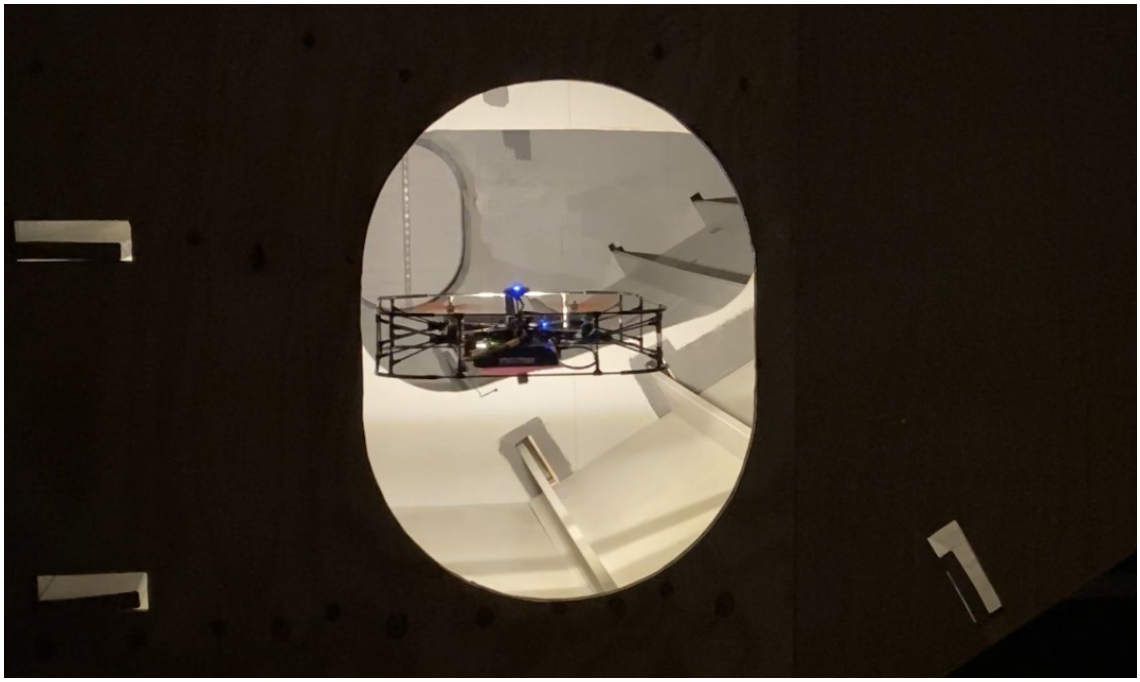


Figure B.1. Our system operating inside the mock-up ballast tank, in complete darkness.

minimize human involvement during inspection of ballast tanks, to eliminate the human risk and to maximize repeatability. The repeatability of such an automated procedure may enable the introduction of defect trend analysis on such critical areas. Due to complex structures within the ballast tanks, ground-based robots are limited by their ability to move on and around the obstacles. Small submarines, like the BlueROV2 from Blue Robotics, could be used for contact-based inspection but are unlikely to provide good results for the visual survey, due to the muddy conditions of the water in the tanks, which greatly reduce visibility. During most —and all of the special— surveys [121], the ballast tanks are empty, and therefore an Unmanned Aerial Vehicle (UAV) is considered to be the most suitable solution for visual inspection task.

To be able to navigate within ballast tanks, as shown in figure B.2, a lightweight accurate 3D localization system is required. Visual Intertial Odometry (VIO) would fit nicely to this description, however the current algorithms —considering optimal conditions for camera sensors— are known to produce at best 1% localization errors which can increase unboundedly [122]. Given poor lighting conditions and long complex pathways in the ballast tanks, the error can be in the 10ths meters. It might come as a natural extension of this line of argument that simultaneous localization and mapping (SLAM) approaches (e.g: ROVIOLI [123] or ORBSLAM [124]) would be a good fit for the task. However, these approaches rely on re-observation (loop-closure [125]), which is not in accordance with automated surveying of ballast tanks. To optimize for exploration, an autonomous drone has to observe every area once and use a quick exit route —usually vertical aligned holes— moving straight up to get out the ballast tank and on the deck to be recharged. Another approach to allow the usage of SLAM would be to conduct a pre-mission to generate a consistent map of the area which can later be used for accurate localization. Given the fact that inspections are usually performed in all tanks of all vessels at most once a year,

this approach would double the load/time to perform inspection.

Following this analysis, the aforementioned approaches are deemed unsuitable for the problem at hand. In this paper we, therefore, propose and test an initial implementation of a Hierarchical VIO-based localization algorithm, which is able to provide absolute —with regards to the vessel body frame— localization, employing 3D structural landmarks, known from design documents. We test the system in a realistic environment (a mock-up model) evaluating both its UAV localization and its defect localization accuracy. The baseline for the evaluation of our system is a proven VIO solution, i.e. ROVIO.

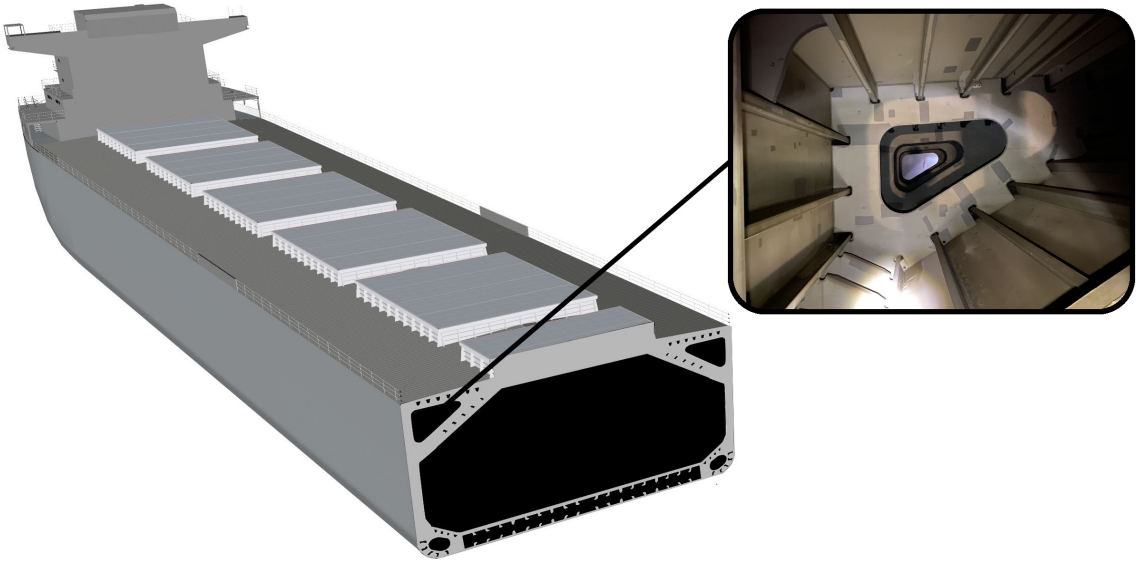


Figure B.2. Location of a top-side water ballast tank in the cross section of a bulk carrier vessel.

B.2 Related work

Entering the confined spaces of vessels is considered a high-risk operation for humans, and the need to use robots has therefore emerged. The most common approaches are magnetic crawlers or small submerged Remotely Operated Vehcicles (ROV). Research on the use of ROVs for vessel inspection has focused on the hull [25], by submerging the robot underwater near the vessel. These approaches usually comprise a SLAM system using a fusion of visual and acoustic sensors. Although small ROVs can physically fit in many of the confined spaces of the vessel, such as the ballast tank, they are not deemed suitable due to both wielding an increasing position error and their water requirement. Though the ROVs might seem as the obvious choice for especially ballast tank inspection, it is important to note that the ballast tanks are only filled with water when the ship is empty, which is often only the case in a very limited time period when the ship is in dock. Crawler and small-legged robots have, to our knowledge, not focused on the localization problem. In fact, the authors of [94] have not implemented a positioning system, and the work in [42] uses external pose referencing. Such a solution is not considered viable for tight, confined spaces such as ballast tanks. In addition, crawlers require human intervention to place and disengage them. This could discourage the adoption of said approaches as they would not provide a solution to the human risk problem. An alternative to pure VIO based solutions is localization based on fusion of data from multiple sensors [126]. The authors of [127]

propose a camera system that can perform localization based on a previously collected LIDAR map. The work shows that the the drift in a VIO system without loop closure can be reduced, by localizing within the prior LIDAR map. The availability of a detailed LIDAR map can not be assumed for our case, and also poses a risk to the localization if the map is not up to date with the actual navigated environment. Other works such as the one in [128] integrates a LIDAR and an IMU to do simultaneous localization and mapping on-board an unmanned aerial vehicle in a GPS denied environment. Localization is performed by using a scan matching approach based on a customized version of the Iterative Closest Point algorithm. Unfortunately the system is intended to only localize itself in a simple 2D map (e.g. office-like), whereas our camera based approach aims at localizing within the full 3D environment. When performing inspection, the location of defects with regards to the body-frame coordinates of the subject is of utmost importance. Therefore, many previous approaches employ pre-existing models during inspections. Even-though there is no extended literature for *body-frame aligned* inspection in confined spaces, outdoor solutions have been reported. The work in [129] requires a 3D CAD model of a bridge as an input to the localization system. Specifically, the system first captures a 3D image and generates a 2D binary projection. Then, it associates it with 2D projections of the CAD model to estimate its absolute position. The requirement for a full 3D model onboard —and the generation of several subsequent projected templates— may lead to exceeding the capabilities of a light weight system, especially for large areas such as the Water Ballast Tank (WBT) of a vessel. Additionally, the method employed doesn't estimate the orientation of the UAV as it assumes a motion always parallel to the inspected area. Since the inspection of defects, requires observation from multiple viewpoints, we opted for a full 6D pose estimate. The work in [130] requires a GPS-enabled pre-mission to generate a reference map of visual landmarks. That enables them to accurately locate themselves using 3D-2D motion estimation. The method employed requires external localization and pre-mission to enable consequent inspections. In other domains (e.g. space exploration [131, 132]) researchers employ FPGAs to accelerate operations. In our approach we extract the necessary absolute landmarks using design documents, therefore, allowing for implementation in a multitude of vessels, in a fast and efficient way.

B.3 System overview

To accurately report the position of defects within a structured confined space, a localization system that rely on the internal structural elements is proposed (figure B.3). To solve this task, the UAV (figure B.1) is equipped with a stereo camera and an IMU, enabling it to use both Visual Inertial Odometry (VIO) and landmark extraction in the form of known structures in the confined space. The UAV uses the VIO as its pose estimation, but experiments show that within the challenging environment of a ballast tank, the accumulated error increases quickly. To mitigate this problem, the developed camera software on-board the UAV extracts structural features —in our case the longitudinals(beams) and web frames in the ballast tank— and compares the estimated positions of the features to the actual known positions within the space. When the pose of the UAV has been accurately estimated, defects within the space can be located with a similar accuracy, i.e., the error of the absolute position of the defects relies only on the error of the absolute localization system, and in our case, the negligible camera projection

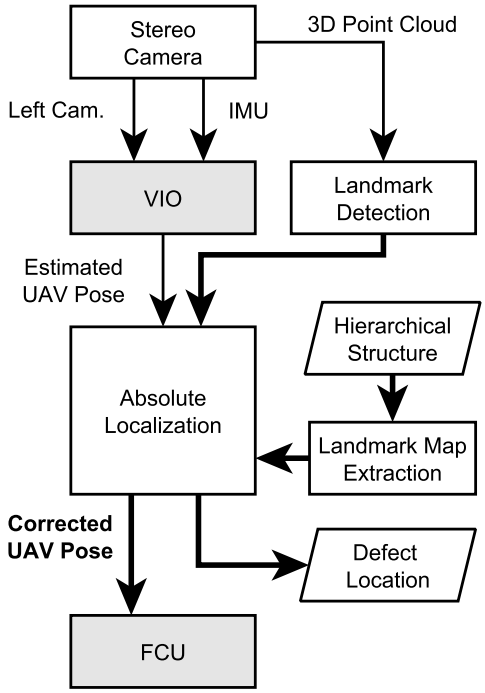


Figure B.3. Absolute localization system based on VIO and hierarchical structure landmarks. The landmarks detected from the UAV's point of view are combined with structural landmarks —extracted from design documents— to provide accurate absolute localization.

error.

A normal use case for our system is as follows: The approximate initial location of the UAV is known —e.g the UAV is placed next to the entrance of the WBT #3. The UAV takes off and detects a manhole, while it localizes itself with respect to it. The UAV then enters the manhole and, consequently, redefines its topological position to be inside the corresponding compartment. The UAV is able to move inside the tank and maintain absolute localization using landmarks, i.e. the corners of the intersections of web-frames with longitudinal strengtheners. When the system passes through another manhole, the localization pipeline is repeated. Given a defect detection system [77] (not included here), the system is capable of providing accurate localization of the observed defects, consistent with the ship body frame. The defects are, in this paper, represented by ArUco markers.

B.3.1 Environment

The developed UAV-based system is aimed to operate in the confined spaces of ballast tanks onboard cargo vessels. To enable rapid development and testing —and due to temporary access restrictions to vessels (see Covid-19)— a mock-up model of a topside ballast tank of a double-sided cargo carrier was designed and produced to simulate the real environment. The mock-up model is at a scale of 2:3, with the manholes being at the original scale. A comparison between the mock-up model and a real top-side ballast tank is shown in figure B.4. Inside a ballast tank one can identify some distinct features as seen in figure B.4c. Among them, the transverse supporting structures (web frames) as well as

(a) Real *(b)* Mock-up model *(c)* Color-overlay

Figure B.4. Photos of a real topside ballast tank (a) of a bulk carrier vessel and its mock-up model (b). We constructed the mock-up at a scale of 2:3 but we maintained the entrances at the same size, to provide realistic conditions. The inside of the mock-up has been treated with paint, so that it assimilates closely the real environment. The longitudinal stiffeners and web frames are color coated in (c).

the longitudinal strengtheners are the most prominent. In this paper, the space between two web frames is defined as a compartment.

B.3.2 Aerial Vehicle

The aerial platform (figure B.5) used in this paper is specifically designed for operating in confined spaces, and utilizes publicly available components, with a focus on minimizing size and weight. It is equipped with a small onboard Linux PC, LatePanda Alpha 868, and a Flight Controller Unit (FCU), Pixhawk 4 Mini —running the open source software PX4 [133], which handles the low-level stability control of the UAV. To sense its surroundings, the UAV is equipped with powerful natural light LEDs and a Mynteye

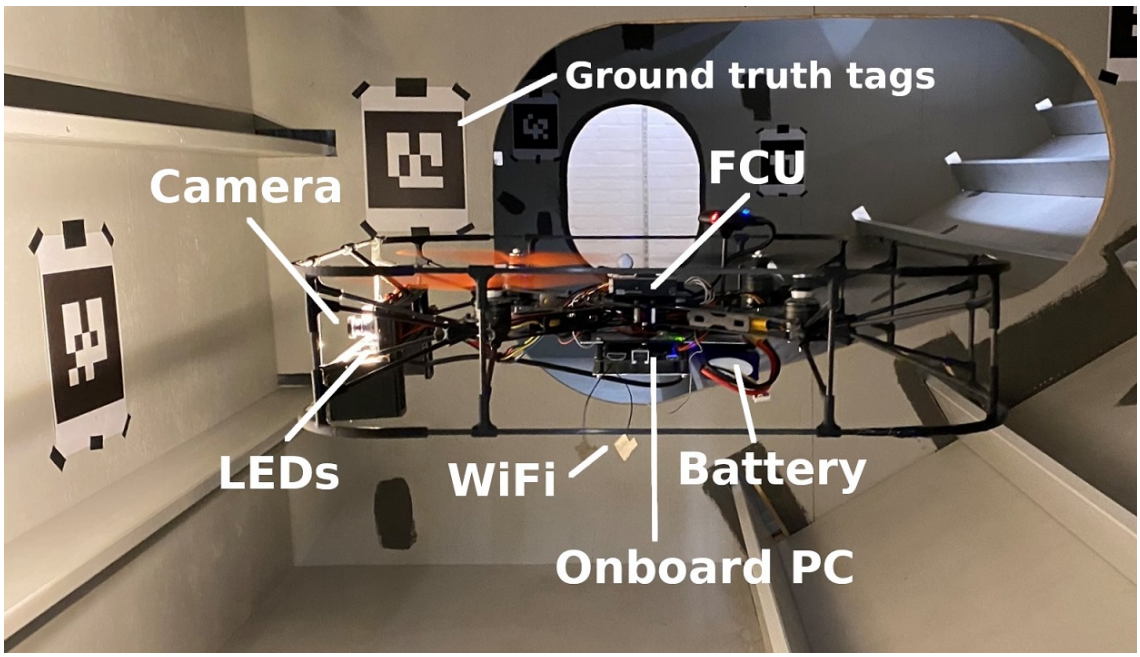


Figure B.5. Our developed UAV system, including stereo cameras, IMU, an onboard flight controller, a processing unit and an adjustable lighting system. N.B.: the ArUco markers on the walls are used for groundtruth and not for localization estimation.

D1000-120 stereo vision camera, which includes a time-synchronized IMU. The camera is equipped with an IR-projector that is continuously toggled with 30 hz. Therefore, we acquire both IR-(with point cloud) and RGB-only images, which are used for VIO. Consequently, this camera setup makes it possible to have both a VIO system running and at the same time generating point clouds for detecting 3D features in the environment. The localization system is executed on the onboard PC which is connected to both the FCU and the stereo camera. The onboard PC transmits pose estimates via serial to the FCU based on the VIO system. To calculate these pose estimates, a fusion of VIO and a association of geometric features with simple prior knowledge of the environment is used, providing a hierarchical localization system.

B.3.3 Visual Inertial Odometry

The UAV uses ROVIO [25] as its base localization system. ROVIO is a Visual Inertial Odometry (VIO) algorithm, which fuses inertial measurements with image data to provide an estimate of the pose of the camera-IMU system, using an iterated extended Kalman filter. ROVIO is selected as our short-term localization system because of its proven robustness and ability to track noncorner-shaped features, such as lines, which is beneficial in scenarios with missing texture. There exist other VIO solutions such as the one in [26] that employ similar algorithms. In general, ROVIO constitutes a robust solution, but its accuracy is challenged by the harsh conditions within the ballast tanks, which results in an increased uncertainty.

B.3.4 Prior Knowledge

Large ships are designed and manufactured with a set number of web-frames, along the lateral axis and a set number of longitudinal stiffeners along the longitudinal axis of the ship, as shown in figure B.4c. The distance between each longitudinal stiffener and frame is also known —described in design documents— and this frame-stiffener construction is employed to guarantee the vessel's integrity. It is also used by inspection personnel as the coordinate system to describe the positions inside the ballast tanks. Therefore, it is important to be able to accurately detect and localize defects with respect to these frames and longitudinals. This prior knowledge is provided to the system in the form of a map of landmarks. This map of landmarks is organized in a hierarchical structure, and includes the 3D location of the intersection points between the web frame and the longitudinals as well as the center of the manholes, with regards to the specific compartment.

B.3.5 Landmark Detection

The estimated pose from the VIO system is corrected by detecting known landmarks, such as the intersection of the web frames and longitudinals, within the ballast tanks of the ship. A point cloud of the environment is created from the disparity map of the camera and the landmarks are thereafter detected by extracting the longitudinal beams and web frames in the point cloud and calculating the location of their intersection, as shown in figure B.6.

The correction is applied every time there is a match between the known and detected landmarks. A detailed description of the landmark detection process can be seen in figure

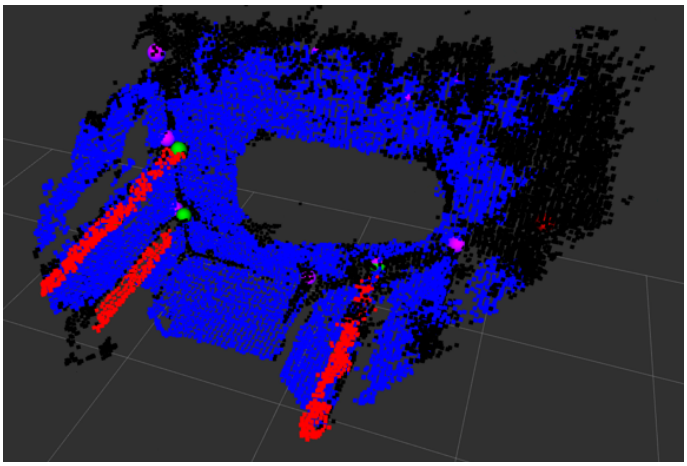


Figure B.6. Illustration of the detected longitudinal stiffeners (red), the intersection points (green), and the known landmarks (purple) in the WBT mock-up model.

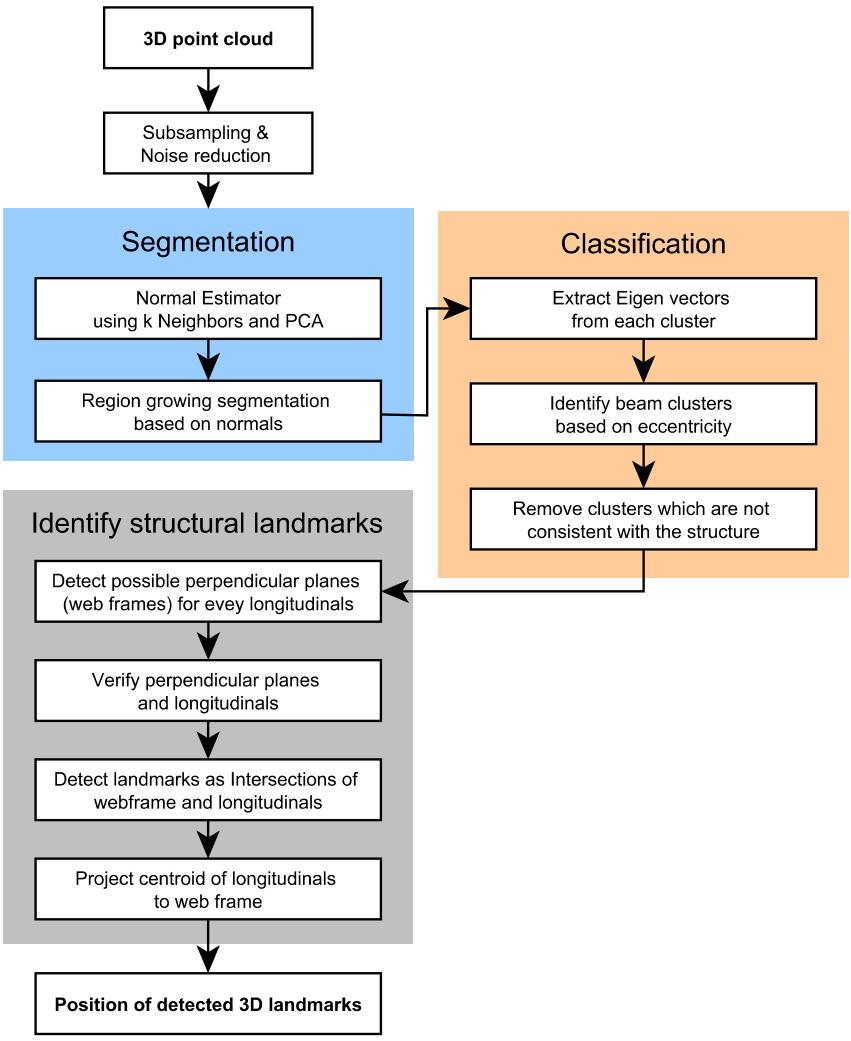


Figure B.7. Algorithm for 3D landmark detection.

B.7. AR landmarks are furthermore detected within the tank with the sole purpose of obtaining a reasonable estimate of the groundtruth pose of the UAV.

B.3.6 Hierarchical structure

A drawing of the structure of the vessel is provided by the owner/operator to the inspection team. Prior to the mission, fixed 3D landmark points were extracted from this drawing. All landmark points are stored in a hierarchical structure, as shown in figure B.8. The top node is a WBT. Each topside WBT consists of several compartments. In each compartment, we include the landmarks as defined in the previous sections. The location of the compartment holes and the location of the landmarks in regard to the compartment they belong to are known. It is possible to request information about a given compartment, which is then used to extract landmarks required for absolute localization. The possible connections between compartments, in the form of manholes, are also included in the structure, and are illustrated in figure B.8 as the arrows between the compartments.

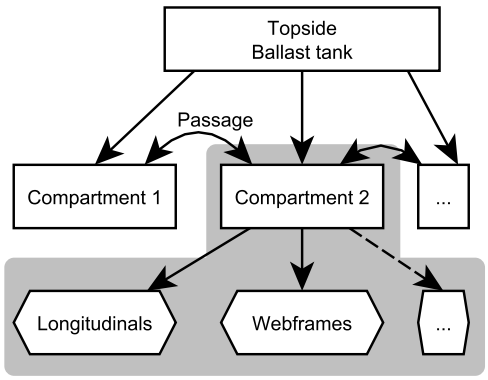


Figure B.8. Hierarchical structure of the Topside WBT. Grey selection represents an example of the information retrieved to be used in absolute localization.

B.3.7 Abs. Localization

Owing to the fact that every passage from compartment to compartment is intentional and actively guided, the system can load the respective landmarks of the "correct" compartment —from the hierarchical structure— to be used for absolute localization. Both the detected and map landmarks have a pose (position and orientation) which can be calculated using the adjacent longitudinal beams. Then, the association among detected and map landmarks can be performed by minimum euclidean distance. A relative "correction" pose can be calculated so that the poses of the detected and map landmarks are aligned. Finally, this correction is added to the VIO estimate to provide the absolute pose of the UAV.

B.4 Experiments

B.4.1 Experimental Setup

All experiments have been conducted within the WBT mock-up model previously shown in figure B.4b. The UAV is launched from a known position outside the WBT and estimates

its relative localization using a VIO system. It is then flown through the first manhole entrance (figure B.1), using position commands on a remote ground station (GS). When inside the WBT, the absolute localization system is activated and corrects the pose of the UAV based on the 3D detected landmarks (longitudinal stiffeners and web frames). ArUco markers are placed inside the tank, and their usage is twofold. On the one hand, they are used to represent defects within the WBT. Once detected, a position estimate of the specific defect/marker is published. This circumvents the usage of a machine learning-based defect detection component and, therefore, allows one to evaluate only the localization accuracy. On the other hand, the markers provide a groundtruth against which both the VIO and the absolute localization will be benchmarked.

B.4.2 Experimental Results

In figure B.9 the trajectories of the position estimates and the groundtruth are illustrated. Both, the results of ROVIO and our method are shown. It can be seen that a standalone

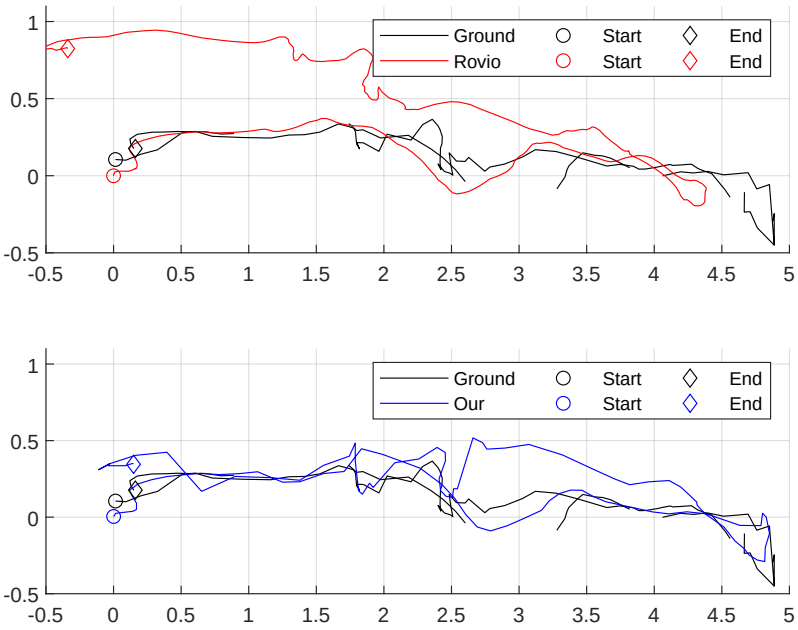


Figure B.9. (X,Y) groundtruth position vs ROVIO estimate (upper) and our approach (lower).

ROVIO solution, implemented on our UAV-setup, is challenged by the harsh conditions in the WBT and the estimation starts to drift away from the groundtruth after 15-30 seconds, whereas our position estimate is corrected every time suitable structural landmarks are detected. The reason of the ROVIO drift can be attributed to several factors, such as lack of features in the dark scene, dust particles in the confined area, and the artificial (on-drone) illumination creating pseudo features which move together with the UAV.

An error in the position estimate of the drone will also affect the ability to localize accurately the defects in the tanks (ArUco makers). Figure B.11 shows the x,y estimated position of the defects and the error, as the Euclidean distance from the measurement to the ground truth. It is clear from the figure that adding the structural landmarks to the position estimate greatly benefits the accurate localization of defects. The hierarchical structure, furthermore, provides increased accuracy without the need for extensive computational resources onboard the drone.

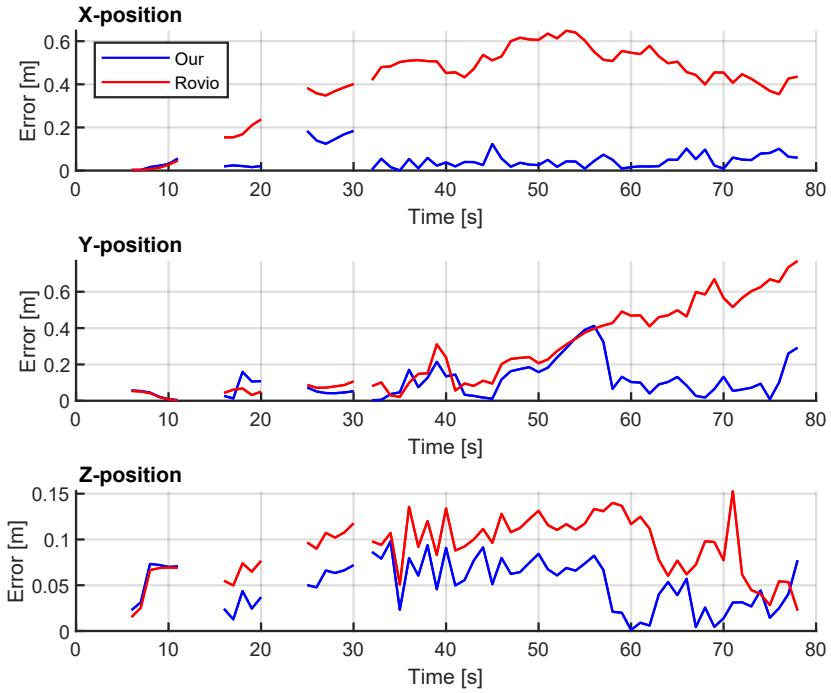


Figure B.10. Position error compared to the groundtruth. Missing data points are due to visual dropouts of groundtruth tags.

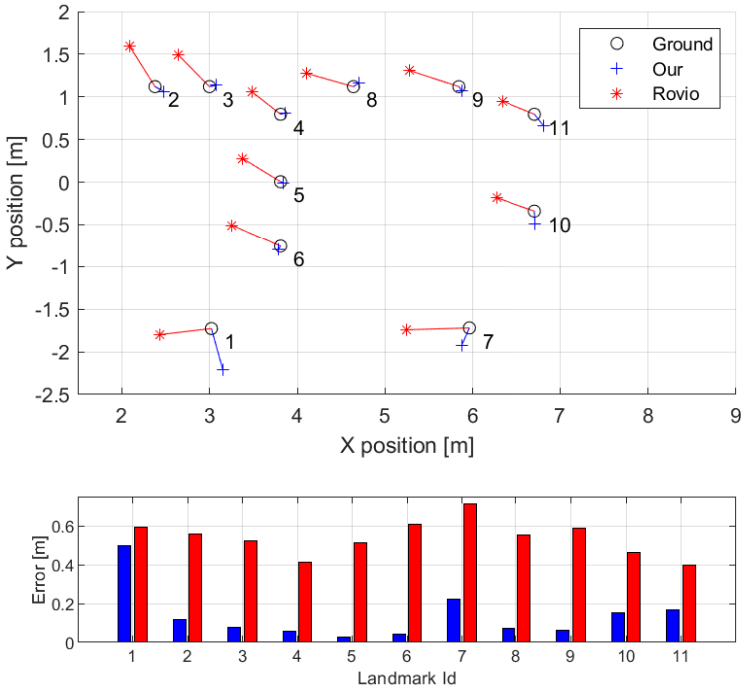


Figure B.11. Defect location estimation vs grountruth (upper) in world coordinates and the resulting error (lower) for ROVIO and our approach.

B.5 Conclusion

Within this work, an absolute localization system, utilizing a hierarchical structure, was proposed for accurate localization of defects in water ballast tanks of marine vessels. A fusion of Visual Inertial Odometry and the detection of known 3D structural landmarks, such as longitudinals and web frames, was conducted to increase the accuracy of the position estimate of the defects. While VIO solutions fail within the dark and confined spaces of the water ballast tanks, our system is capable of accurately localizing defects, requiring only a high-level human understanding of the environment. Our experiments conducted in a mock-up model of a topside water ballast tank, proved that the localization system was able to accurately provide better positions of defects located within the confined space compared to a stand-alone VIO system. In our future work we will integrate the updates from visual-inertial ego-motion and landmark-based pose extraction, in a tightly-coupled state estimation. Even though we did not experience many erroneous 3d landmark extractions in our environment, in the future, we will investigate the performance in other types of water ballast tanks, as well as the resiliency of landmarks and data association.

Chapter 4

Absolute Localization with Deep Learning landmarks

In the previous chapter, we investigated a method for localizing using custom structural landmarks extracted from point clouds. There are a series of benefits to this method, but one of its weaknesses is the need to create custom landmark/feature descriptors for the environment. In this chapter, we introduce a setup for absolute localization in known confined spaces, using Deep Learning based landmarks, which is applicable to a wider range of cases. The focus will be on using 3D point clouds from a camera-based ToF sensor, for absolute pose estimation. However, the most common approaches for point cloud registration will first be considered.

4.1 Iterative Closest Point (ICP)

The Iterative Closest Point (ICP) algorithm is a popular method for rigid registering of two point clouds. The goal of ICP is to find the transformation that minimizes the distance between the corresponding points in two point clouds. The most common approach for the ICP algorithm is to create a correspondence set by finding the closest distance between the points from a source point cloud P, to a target point cloud Q. This is done for all the points in the cloud, and a transformation is estimated based on the minmization of an objective function defined over the correspondence set. The ICP algorithm then works iteratively, with each iteration improving the alignment of the two point clouds, until a convergence criterion is met or a maximum number of iterations is reached. The convergence criterion commonly used is the maximum RMS error of the correspondence points.

One of the drawbacks of the ICP algorithm is slow convergence due to its linear convergence rate [134]. Moreover, ICP needs an initial starting position and can also converge to a local minima, as shown in the example in Figure 4.1. Using classic ICP as the only registration solution for 3D point clouds could, therefore, potentially give a lot of false registrations and in the same time be computationally heavy. Much of this potential error comes from the data association step in the algorithm in which the correspondence set is generated. If the points were paired based on some uniqueness of each point, this could solve some of the incorrect correspondences. This uniqueness can be achieved by employing feature descriptors.

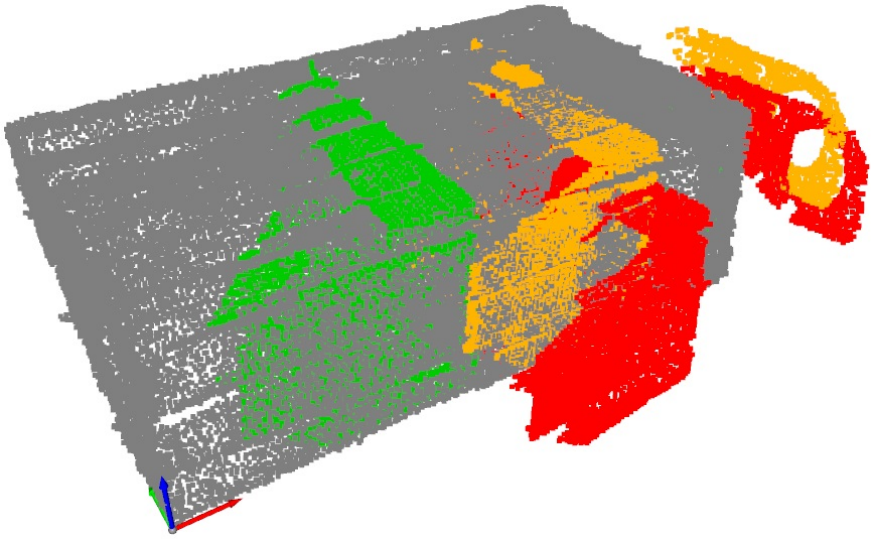


Figure 4.1. Illustration of a sensor scan (red) registered to a map (grey) using point-2-point ICP, with the registration result shown in yellow. The structure of the environment can cause ICP to converge to an incorrect local minima. The correct registration is shown in green.

4.2 Feature descriptors

A common way to describe the structure in 3D point clouds is to use 3D feature descriptors. These descriptors use various methods, where a popular choice for a classical feature descriptor is Fast Point Feature Histograms (FPFH)[135]. FPFH uses histograms to represent the local geometric structure surrounding a single point in a 3D point cloud. To do this, first the surface normals of the surrounding points are calculated and the angles between the normals are extracted. The angles are then binned into histograms, with a fixed number of bins. One of the advantages of FPFH, according to its authors, is its ability to handle noise and changes in viewpoint. Since the descriptor is based on the distribution of normal angles around a point, FPFH is not sensitive to small variations in the positions of the points. Furthermore, by using FPFH to represent the local geometric structure, the descriptor can capture the shape of a local area, rather than being affected by small variations of the surface, e.g. from noise. However, these abilities also depend on how well the normals are calculated for each point and, thus, the descriptor can require a bit of manual tuning of its input data.

In recent years, deep learning-based methods for 3D point description have been investigated with increasing success. One of the current powerful feature descriptors is 3DSmoothNet[136] that applies a novel smoothed density value (SDV) voxelization preparation step on the input data. The advantage of SDV is that it reduces the sparsity of the input voxel grid. The output of the SDV is then used as input to a siamese deep learning architecture. The complete network is trained on the *3D Match* dataset [137], which consists of aligned point cloud data from indoor environments such as offices, hotels, and homes. The ML model used in the experiments in this chapter is trained on this dataset and should, therefore, be expected that the descriptors' ability to represent the environment is diminished due to domain change. However, the results of the articles in this chapter show that 3DSmoothNet performed well, albeit computationally resource intensive. A large part of the computational cost arises from the data preparation step of

the SDV calculation.

Moving towards a more computationally efficient pipeline, we employ the state-of-the-art Fully Convolutional Geometric Features (FCGF)[138]. The FCGF feature descriptor eliminates the need for a data preparation step and works directly on the 3D point cloud data, hence its improved computational efficiency. Furthermore FCGF is based on sparse tensors and sparse convolutions, which adds to its efficiency. It utilizes a UNet structure with skipped connections and residual blocks to extract sparse fully-convolutional features. To counteract the loss of accuracy in benefit of computational efficiency, FCGF employs negative sample training and specifically introduces novel contrastive losses, maintaining both high efficiency and high performance during training. The use of FCGF as a suitable feature descriptor for ballast tanks was investigated in **article D**. The FCGF model used in the article was also trained on the *3D Match* dataset. FCGF did not perform as accurately as 3DSmoothNet but was 6.7 times faster in the experiments conducted in the article. The lack of accuracy can be overcome by employing registration algorithms that are robust in high outlier count. This will be the subject of the next section.

4.3 Correspondence based Registration

Assuming features extracted from two overlapping point clouds, theoretically a set of matching features should exsist. The data association between the feature points can be done using K-Nearest Neighbors (KNN) in the feature space to find the corresponding 3D points. In **article C** the KNN correspondence search is performed on the Central Processing Unit (CPU), while in **article D** the search is performed on the GPU using Facebook AI Similarity Search (FAISS)[139]. FAISS was originally developed to quickly search through large amounts of multimedia documents by harvesting the parallelizable compute architecture of the GPUs. It is built around an index type that stores a set of vectors and provides a function to search through a list based on Euclidean distance (L2) in the feature space. The results show a noticeable time improvement utilizing the GPU-based correspondance search from FAISS compared to the standard CPU-based version in **article C**.

With a list of corresponding points, a registration algorithm can be used to find the optimal alignment of two point clouds, in the presence of outliers. A common way to do this is to use RANdom SAmple Consensus (RANSAC). RANSAC randomly selects a subset of the 3D points from the feature descriptor correspondences and estimates the transformation matrix that aligns these points. This random selection can be repeated up to a predefined maximum number of times. For each random sampling, the algorithm applies the estimated transformation matrix to the source point cloud and validates the transformation by considering the convergence criteria of all the points in the point cloud. The process finishes either by reaching the maximum number of iterations or by satisfying the convergence criterion. The convergence criteria for a successful registration are, in the following articles, based on a threshold of required inliers in the corresponding point clouds (as well as the maximum number of iterations of RANSAC). Depending on the number of features and iterations required, RANSAC can be computationally inefficient. Furthermore, a large number of outliers in the correspondences can also affect RANSACs ability to converge to the correct solution[140].

To counteract this problem, a newer method, called Teaser++[141], has been proposed in the literature as a fast and robust approach to register two sets of 3D points in the presence of a large number of outliers, although for object registration rather than localization. This method has been the preferred choice for the registration algorithm in **article C** and D. Incorporating a feature descriptor, correspondence search, and a registration algorithm into a 3D registration pipeline, a scan from the ToF point cloud sensor can be registered to the map of the environment. The transformation of the scan into the map frame provides a pose estimate of the sensor, and by extension also of the robot, within the given map. This is the absolute pose estimate of the robot. However, this pose estimate can still be computationally intensive and result in slow update rates on performance limited platforms. Therefore, a relative pose estimate can be used between absolute pose updates. By fusing the slow absolute pose with a faster relative pose (VIO) in an Extended Kalman Filter (EKF), a fast absolute pose estimate can be obtained. Figure 4.2 illustrates how the 3D

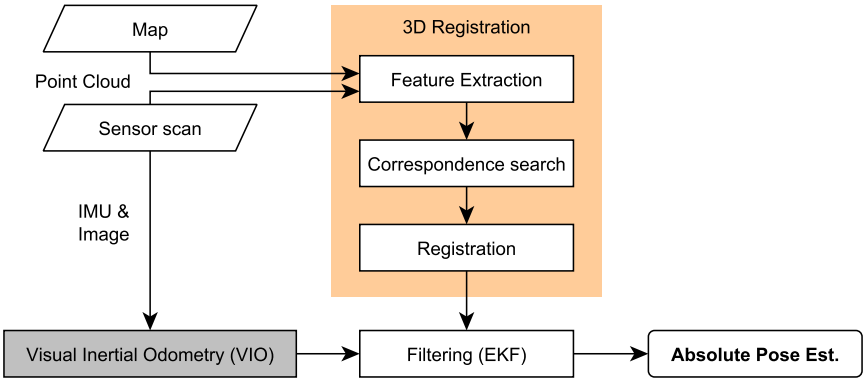
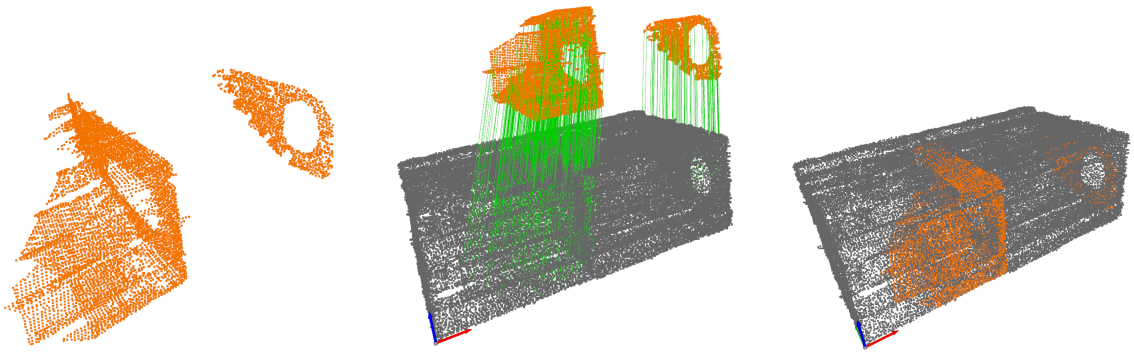


Figure 4.2. Illustration of the high-level absolute pose estimation by fusion of both 3D feature based registration and visual inertial odometry.

registration is fused with a Visual Interial Odometry estimate in an EKF, to provide the fast absolute pose estimate. The high-level representation of the 3D registration pipeline containing the feature descriptor, correspondence search and registration, can be seen as the orange *3D Registration*-box in the figure. An example of how a point cloud scan is registered to the environment map is shown in Figure 4.3, which in this case is based on the ballast tank mock-up model. The green lines indicate where features from the orange sensor scan are matched to the map, marked as gray points.

4.4 Summary

The motivation behind this research section was to eliminate the need for custom feature extraction and investigate the possibility of 3D Deep Learning-based feature descriptors for absolute localization in known confined spaces. **Article C** compared FPFH and 3D smoothnet in combination with ICP and Teaser++ registration. The findings showed that FPFH as a feature descriptor for the highly self-similar environment of the ballast tank did not perform well enough —in terms of accuracy— and, therefore, was one of the main components to be replaced on the follow-up work and corresponding article. 3DSmoothNet proved to be a good feature descriptor for the environment and was able to provide absolute pose updates with decent accuracy. The absolute pose estimate was



(a) Sensor scan *(b)* Feature Correspondences *(c)* Registration result

Figure 4.3. Example of a scan to map registration from a water ballast tank mockup model, where the orange orange point cloud is the scan from the point cloud sensor and the gray points indicate the environment map. The green lines indicate the matching correspondences between the two point clouds.

coupled with a relative VIO estimate in an extended Kalman filter to provide a fast and accurate absolute pose estimate. In **article D**, the newer and faster FCGF model was applied and compared to the previously investigated 3DSmoothNet. FCGF was a faster method than 3DSmoothNet due to its sparse tensor representation of the 3D space. In this work, both 3DSmoothNet and FCGF were executed on the GPU and the article, furthermore, introduced an improvement to the correspondence search by utilizing FAISS KNN —also implemented on the GPU. These improvements resulted in the localization pipeline being able to run on live data from the sensors in adequately high frequency for onboard online operations.

There is some similarity between the two articles in this section. They both utilize feature descriptors to register a scan point cloud to a map, to provide an absolute pose estimate in a map frame. However, there are still important key differences that need to be highlighted. In the first published **article C**, all computations for the localization system are performed offline on the CPU and the system was tested on a collected dataset. In the subsequent **article D** it was clear that utilizing a GPU for the feature descriptor could significantly boost the computational performance of the feature descriptors based on deep learning. The goal was therefore to introduce the GPU as a significant aid for the aforementioned 3D registration part of the localization pipeline. The findings of both articles show how using the GPU can significantly boost the performance of these kind of point-cloud-based localization systems, and using a carefully selected pipeline, the system can run live in real environments.

Article C

Absolute Localization in Confined Spaces using Deep Geometric Features

Rune Y. Brogaard[1], Ole Ravn[2], Evangelos Boukas[2]

[1]FORCE Technology, Park Allé 345, Brøndby, Denmark

[2]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

Abstract:

When operating in dark and confined spaces, the capacity of the robots to localize in an absolute reference frame is of utmost importance. This letter presents an absolute localization system, using deep 3D landmarks, for known confined spaces. The system estimates the robot's relative localization using visual inertial odometry. Local deep 3D landmarks are extracted from the robot's view. Similar 3D landmarks are, also, extracted from the prior map, which are then registered with the local landmarks to provide absolute localization via an Extended Kalman Filter. To the best of our knowledge, deep 3D feature registration has not been used before for absolute localization. The proposed localization system is tested within representative application area —i.e. a structured, confined space— and our results indicate greater accuracy and lower processing time when compared to mainstream 3D registration approaches.

C.1 Introduction

Localizing within industrial dark and confined spaces can be a challenging task in robot applications. Structured areas, such as industrial environments, often contain featureless surfaces hindering visual tracking or recognition. Visual localization systems are known to produce at best 1% error [122], [142] which is also unbounded without some prior knowledge or re-observation. Repetitive patterns —of both visual and structural nature— can make loop closure, and in extension absolute localization, prone to errors. Furthermore, occlusions can diminish the accuracy of VIO systems, while low light conditions limit the number of reliable visual landmarks. Additionally, environmental attributes, such as dust, can introduce noise to sensors and, in consequence, to the estimation. Using structural

features from confined spaces for absolute localization has been previously investigated in [2], however, the landmarks were custom designed to represent use of case-specific structural attributes. The authors of [143] use the VIO pose as a priori for the ICP registration, and add a further validation of the registration result. However, they perform registration directly on the original points without extracting distinct features.

Another approach investigated by the authors in [14], is to fuse positions from a pre-installed Ultra Wide Band(UWB) satellite system with a VIO pose estimate. Although this method could be suitable for warehouses, it is not possible to equip all the relevant GNSS-denied confined spaces with UWB systems. Another application area for GNSS-denied absolute localization is the robotics space exploration [144, 145]. A different approach could be the referencing of online data to offline 3D models (stemming either from CAD or from previous stereo reconstruction[146, 147]).

Absolute localization using geometric 3D features has become an interesting research topic due to the rise of better feature detection [136], [148] and robust registration approaches [140], [141], [149]. The aforementioned novel registration approaches have been applied to the problem of object pose estimation, rather than robot localization. To the best of our knowledge, this is the first work employing deep 3D feature registration for absolute localization in industrial confined spaces.

In this letter, we propose an absolute localization system for confined spaces using deep 3D geometric features extracted from point clouds. Extracting geometric features from both the pre-existing map and the current field of view of the robot, we find the relative pose between the two point clouds to obtain an absolute pose in the map frame. Due to the computational cost of the point cloud registration, we employ VIO —which provides high-frequency relative localization— and couple it with the absolute pose at a lower frequency, via an Extended Kalman Filter.

Using state of the art deep geometric feature descriptors, on both the robot's field of view and the already available CAD models of confined spaces, we can extract better point correspondences between the clouds. Combining these correspondences with a fast and outlier-robust registration algorithm results in better registration performance compared to previous methods. Furthermore, fusing a fast robust VIO relative localization system with the registration, gives rise to a reliable absolute localization system even in highly repetitive structural environments.

The rest of the letter will describe the system architecture, the experiments conducted in a real-world confined area, and the performance of the system with regards to position accuracy and computational time.

C.2 System description

The proposed localization system, figure C.1, consists of two modules, the first of which is pose estimation based on 3D point cloud registration. The second module contains the filter, which rejects 3D registrations with sudden motion changes, and fuses the 3D registered pose estimations with VIO estimates to increase the update rate of the resulting absolute pose estimation.

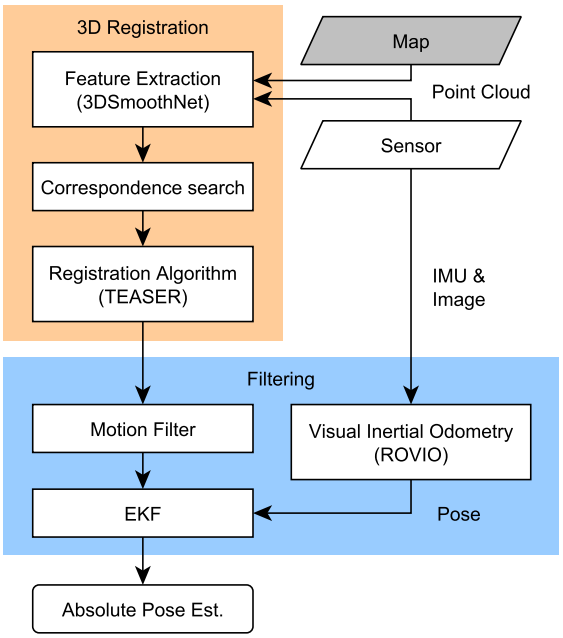


Figure C.1. The pipeline of the proposed absolute localization system. The main parts of the system are the Feature extraction (using 3DSmoothNet), robust 3D point registration (using TEASER++), and the Extended Kalman Filter which fuses the pose estimate from the registration algorithm with a visual-inertial pose.

To extract features in both the map and the sensor point cloud, we employ a 3D deep convolutional neural network-based algorithm, i.e: 3DSmoothNet [136], which comprises a two-part pipeline. The first part is a voxelized smoothed density value (SDV) representation, which is computed per interest point and aligned to the local reference frame (LRF) to achieve rotation invariance. The second part is the siamese deep learning architecture which receives as input the normalized 3D SDV voxel grid. The result of the 3DSmoothNet pipeline is a feature descriptor for the given interest points in the point cloud. The interest points are selected randomly —approximately 50% of all points in the map and 25% of the sensor point cloud. By randomly selecting points in both clouds, we minimize the computational load while maintaining a high probability of acquiring correspondences. The feature descriptors of the interest points are organized in k-d trees to allow the efficient correspondences search between the map and sensor point cloud.

The noisy corresponding points are fed into a robust registration algorithm, i.e: TEASER++ [140] which calculates the transformation between the map and sensor cloud. In an ideal case, where no outliers exist among the correspondences, the registration can be defined as a nonlinear least squares solution, as follows:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left\| \mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t \right\|^2 \tag{C.1}$$

, where the minimization is performed over the scale s, the rotation $\mathbf{R}$, and the translation t. Due to the metric environment of the use case, a scale of 1 can be assumed and, therefore, no additional computational resources will be spent on estimating the scale. The 3D correspondence points are notated as $\mathbf{a}$ and $\mathbf{b}$, where (a_i, b_i) represent a correspondence

pair between the map and sensor point cloud features and N represents the total number of corresponding points. Due to the noisy measurements of the point cloud reconstruction, a Gaussian noise with isotropic covariance described by σ^2 is included. However, for most real world cases, correspondences with zero outliers cannot be safely assumed [140] and therefore the robust registration can be performed using a Truncated Least Squares function C.2, as follows:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \min\left(\frac{1}{\beta_i^2}\left\|\mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t\right\|^2, \bar{c}^2\right) \tag{C.2}$$

Equation C.2 describes a least squares solution of measurements with small residuals ($\leq \bar{c}^2$), where β_i is a given bound of the noise. This noise bound is set as the maximum error expected from an inlier or 3σ. Large residuals ($> \bar{c}^2$) are discarded. In our experimental setup, $\bar{c}$ is set to 1.

To simplify the solution, the rotation and translation are decoupled in TEASER++, as expressed by equations C.3 and C.4 respectively.

$$\hat{\mathbf{R}} = \arg\min_{\mathbf{R}\in SO(3)} \sum_{k=1}^{K} \min\left(\frac{\left\|\bar{\mathbf{b}}_k - \hat{s}\mathbf{R}\bar{\mathbf{a}}_k\right\|}{\delta_k^2}, \bar{c}^2\right) \tag{C.3}$$

Firstly, to estimate $\hat{\mathbf{R}}$, we want to minimize the distance between corresponding points expressed as $\bar{\mathbf{b}}_k$ and $\bar{\mathbf{a}}_k$, with a bounded noise δ_k. Using the estimated rotation, the translation can then be determined by equation C.4. The translation is acquired component-wise, i.e., the entries t_1, t_2, t_3 of $\mathbf{t}$ are computed independently. Large residuals ($> \bar{c}^2$) are, as in the previous equations, discarded:

$$\hat{t}_j = \arg\min_{t_j} \sum_{i=1}^{N} \min\left(\frac{(t_j - \left[\mathbf{b}_i - \hat{s}\hat{\mathbf{R}}\mathbf{a}_i\right]_j)^2}{\beta_i^2}, \bar{c}^2\right) \tag{C.4}$$

An analytic derivation of the aforementioned formulation can be found in [140]. Via the registration, we obtain the pose of the robot in the map. The registration assumes no knowledge about previous poses and a simple motion filter and an extended Kalman filter are introduced to improve the pose estimate. To obtain a better estimate, the pose from the 3D registration is evaluated by a motion filter, which rejects movements with a velocity above 0.3 m/s. The filtered pose is then used in an Extended Kalman Filter along with the VIO pose. The result of the Kalman Filter is an absolute pose estimate in the map.

C.3 Experimental Setup

The proposed localization system has been tested within a representative structured and confined space, specifically a mock-up model of a topside water ballast tank (WBT) onboard a marine cargo vessel. The structure of these confined spaces are often known. The localization system can use this information in the form of a point cloud extracted from the CAD drawings, which we consider as the *map*. The environment, or the *map*, the UAV has to localize itself in, can be seen in figure C.2. The experimental Unmanned Aerial

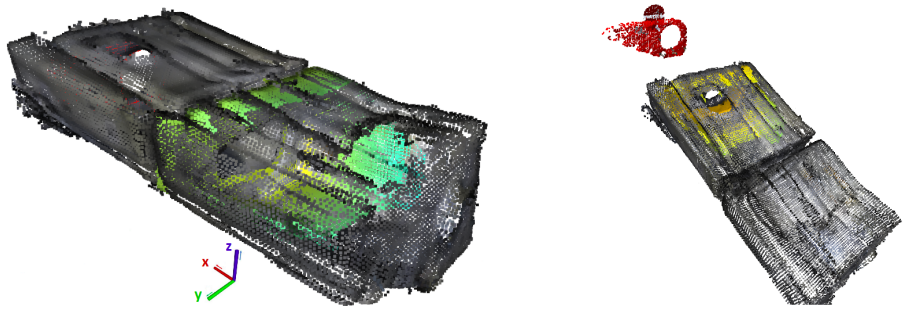


Figure C.2. Illustration of a correct(left) and wrong(right) match with the point clouds used in the experiments, where the map is shown in grey and the sensor points in colour. The geometric similarity of repeated structured environments can cause errors when matching without prior location information. These wrong matches are avoided using simple filtering techniques.

Vechicle (UAV) was equipped with an RGB-D camera, which generates RGB images and 3D point clouds at a frequency of 15-24 Hz. To estimate the position of the UAV, the point cloud originating from the sensor is matched to the map point cloud. The registration is based on TEASER++ [140] and uses features generated by 3DSmoothNet [136] which utilizes a deep learning architecture to extract 3D features from point clouds. To compare the performance of our system, two other setups were tested. The first setup is based on the extraction of FPFH [135] features, and registration using TEASER++. To maximize the quality of the FPFH setup, features were calculated for all points in both the map and the sensor point cloud. The pose estimate from each solution was compared to a groundtruth, which was estimated using ArUco markers mounted inside the mockup-model. From the position error in figure C.3 it can be seen that the FPFH solution has large errors on all three axis and, therefore, cannot register correctly the sensor and map point clouds. To get better matches, we therefore introduce 3DSmoothNet feature descriptors, while maintaining TEASER as the registration algorithm (shown as the red line in the figure C.3).

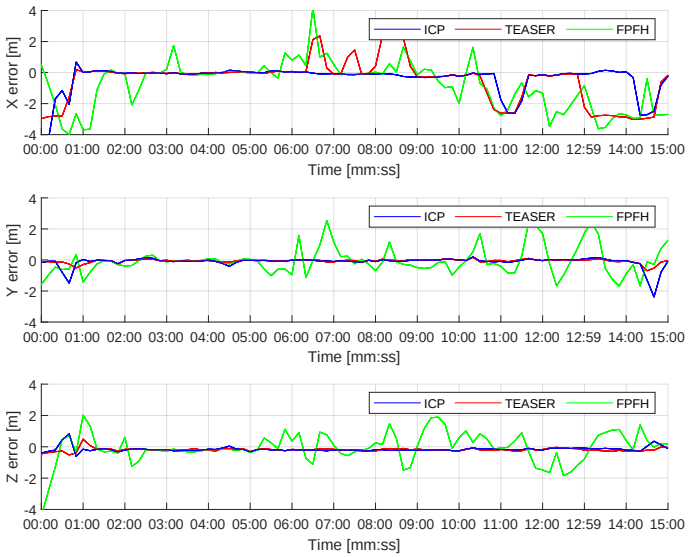


Figure C.3. Position error for tested methods: ICP registration using: 3DSmoothNet features; TEASER++ using 3DSmoothNet features; and TEASER using FPFH features.

As the last comparison setup, ICP registration [150] was used based on the features from 3DSmoothNet(3DSN) and the points in map and sensor clouds. The registration for all tests was performed at 0.1 Hz. From figure C.3 it can be seen that both 3DSmoothNet(3DSN) with TEASER++ and the ICP test show sudden errors of around ±2 meters on the x-axis. Due to the similarity between the two compartments of the WBT, the sensor cloud is being matched to the wrong compartment, which results in large jumps in the matched pose. An example of a correct and wrong match is shown in figure C.2. These jumps are rejected by a motion filter which rejects updates with a velocity larger than 0.3 m/s, before the estimated pose from TEASER is published to the Extended Kalman Filter, as shown in figure C.1. The extended Kalman filter fuses the pose from TEASER++ with a VIO estimate from ROVIO [25] using a ROS implementation of a generic Kalman Filter [151]. The VIO pose has the benefit of a fast update rate, but will tend to drift over time. By fusing the accurate but slow pose from the 3D registration with the fast VIO estimate,the final accurate absolute pose can be acquired with the high update rate from the VIO system. The result of the entire localization pipeline is shown in figure C.4, where it is clear that the position error for ROVIO increases, while the filtered EKF pose error is small. The mean and standard deviation of the tested methods can be seen in table C.1.

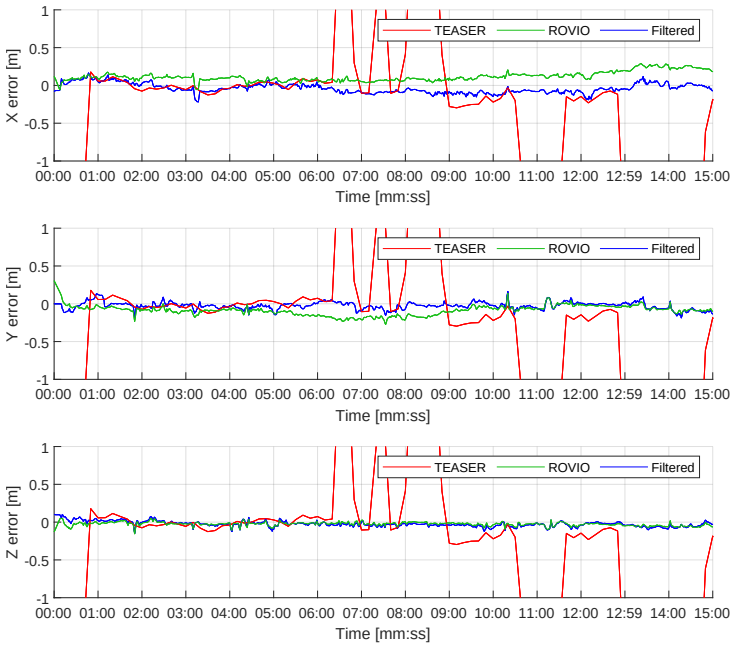


Figure C.4. The position error of our filtered system (Notated as **Filtered**) compared to the standalone TEASER++ using 3DSmoothNet (Notated as **TEASER**), and ROVIO pose estimate (Notated as **ROVIO**). Because of ROVIO's relative localization, an increasing error is visible on the x axis, but due to little movement ($\pm 1m$) in the experiment on the y and z axis, the drift in ROVIO is limited on these two axes.

	3DSN+ICP	FPFH+TEASER	3DSN+TEASER	ROVIO	Our.
Mean	0.715	1.842	1.094	0.1724	0.102
Std.	1.138	1.304	1.153	0.0564	0.050

Table C.1. The mean and standard deviation(Std.) of the euclidean position error, in meters, of the tested methods: ICP with 3DSmoothNet features, TEASER++ with FPFH, TEASER++ with 3DSmoothNet, ROVIO, and our EKF filtered absolute localization solution.

C.4 Computational time

An important aspect for all localization systems is the computational time required to obtain a pose estimate. The computational time for our proposed approach(3DSN+TEASER) is illustrated in figure C.5 and is compared to the FPFH and ICP setups. To have a fair comparison, all tests were performed on a CPU, and no GPU was used for 3DSmoothNet, despite the fact that even better performance would be expected by using a GPU. In figure C.3 it can be seen that the ICP registration with 3DSmoothNet features has fewer outliers than 3DSN+TEASER++, but this accuracy comes with a computational cost. The computational costs of the three methods are shown in figure C.5, where it can be seen that the combination of 3DSmoothNet(3DSN) and Teaser is more than twice as fast as the ICP solution. It is worth noticing that the correspondence search is significantly slower with the FPFH solution since it's based on all points in the point cloud. This was necessary to get the best registrations with TEASER++ and, therefore, it provides the lowest possible error for the 3D point matches.

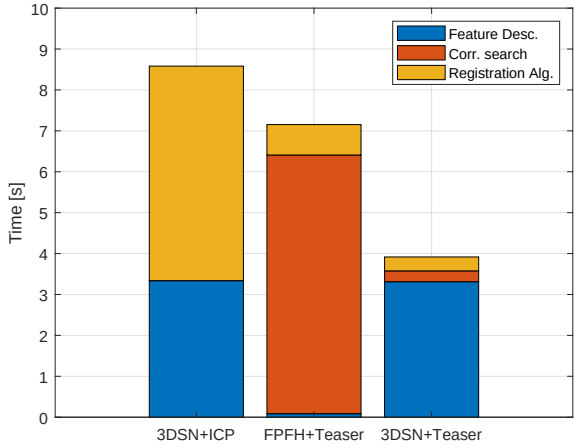


Figure C.5. The average computational time for each element in the 3D registration methods; (i) ICP registration using 3DSmoothNet features (3DSN+ICP), (ii) TEASER++ registration using FPFH features (FPFH+TEASER), (iii) TEASER++ using 3DSmoothNet Features (3DSN+TEASER).

C.5 Conclusion

In this letter we have proposed an absolute localization system for confined spaces using VIO and deep geometric features, which is able to accurately estimate the position of an UAV in a confined space. The experimental results in real environment indicate that our absolute localization system outperforms standard feature detection approaches such as FPFH, and lowers the computational time from scan to registration, compared to previous methods. Furthermore, fusing the pose of the registration with a VIO estimate in an extended Kalman Filter increases the robustness in similar structural environments and can increase the update-rate of the absolute pose estimation. Future improvements to the system can be achieved by the inclusion of faster deep geometric feature descriptors, which will further robustify the 3D point cloud registration.

Article D

GPU-accelerated Localization in Confined Spaces using Deep Geometric Features

Rune Y. Brogaard[1], Ole Ravn[2], Evangelos Boukas[2]

[1]FORCE Technology, Park Allé 345, Brøndby, Denmark

[2]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

Abstract:

Navigating within dark and confined spaces requires robotic platforms to utilize accurate and reliable localization systems to operate safely and unattended. This paper presents an absolute localization system, for known confined spaces, using state-of-the-art 3D pointcloud descriptors. Local geometric features are extracted from a known map and registered to matching features visible in the robots' field of view. The 3D registrations are motion-filtered and fused with a visual inertial odometry estimate in an extended Kalman filter, which returns a fast and accurate absolute pose estimate. The proposed localization system is tested with different deep learning feature descriptors in a structured confined space, and our results indicate greater accuracy and lower processing time when compared to mainstream 3D registration approaches.

D.1 Introduction

Absolute localization within dark and confined spaces, using features from the environment, can be a difficult task in robot applications. Industrial spaces, not designed for everyday human interaction, can contain featureless surfaces, limiting pure visual tracking or recognition algorithms. Visual Inertial localization systems are prone to unbounded errors and are known to produce at best 1% error [122], [142]. Repetitive patterns, of both visual and geometric nature, can be used for loop closure, but in environments with structural similarities, they are prone to errors.Visual Inertial Odometry(VIO) systems also depended heavily on sufficient light conditions and furthermore tend to drift significantly in the case of dust, smoke or similar noise in the environment [152]. Similarly, highly performing visual methods for loop closure and simultaneous localization and mapping [153, 154, 155],

are also not good candidates when operating in the absence of light or in foggy/dusty environments.

The employment of custom, manually designed, 3D geometric features for absolute localization was described in [2], where 3D —CAD derived, case-specific— structural elements were detected in point clouds. Exploiting the 3D geometry to provide accurate mapping has been investigated in [143] where VIO was used as the first step of an ICP-based localization and mapping algorithm. In that case, the whole pointclouds were used and the feature extraction was not investigated.

If the confined space consists only of large open accessible areas, fixed positioning systems such as UWB beacons, can be installed and used as a position estimate. This approach was investigated in [14], where the authors fuse positions from a preinstalled Ultra Wide Band (UWB) satellite system with a VIO pose estimate. While interesting for fixed —frequently explored— areas such as workshops and warehouses, it is clear that for most cases —such as new exploration or in-frequent operation— the use of UWB beacons as a counter measure to GNNS-denied areas is impractical. Extraterrestrial planetary robotic exploration is also an application where the GNSS-denied is crucial. In [144, 145] 3D landmarks, such as large boulders and craters —observed both in rover and orbital images— were used to localize a rover in cartographic coordinates.

Due to the rise of fast and accurate 3D descriptors, such as [136], [148], [138] and robust registration approaches [140], [141], [149], absolute localization on 3D point clouds has the potential to become a well-performing approach. Inspired by these latest advancements, in this paper, we: a) propose the absolute localization of an autonomous system using deep geometric feature matching and b) benchmark the two most promising deep geometric feature description algorithms. The recently proposed deep geometric feature descriptors allow us to produce better correspondences between the online robot 3D reconstructed pointclouds and the previously extracted model-based 3D pointcloud of the confined area.

The correspondences are further evaluated and pruned by a state-of-the-art registration algorithm which is both fast and robust against a large percentage of outliers, yielding better overall performance than previous 3D-based localization systems. Finally, the fusion of the periodic registration with a online high frequency VIO relative localization results in an absolute localization system which is reliably performing even in cases of highly repetitive structural environments. We have explored the idea of deep geometric features for absolute localization in [3] using 3DSmoothNet. In this paper we propose an updated pipeline which, with the inclusion of a GPU-based correspondence search and a substantially faster deep feature descriptor, allows for computational times suitable for online robot execution, which was previously unfeasible. The following sections will describe the system architecture, the experiments conducted in a real-world confined area, and the performance of the system with regards to position accuracy and computational time, using the feature descriptors; Fast Point Feature Histograms (FPFH), 3DSmoothNet and Fully Convolutional Geometric Features (FCGF).

D.2 System description

The absolute localization system, shown in figure D.1, is divided into three components; a 3D registration module, a visual inertial odometry system, and a filtering module. In the 3D registration module, the point cloud generated by the sensor is registered to a map of the environment, to generate the first pose estimate. Due to the occurrence of incorrect registrations, the pose is fed to a motion filter to reject sudden jumps in the pose, which may occur due to these outliers. Parallel to the 3D registration, a fast VIO system estimates a relative pose based on IMU and image data from the same sensor. The VIO and motion filtered registration poses are fused in an Extended Kalman Filter, to calculate a reliable and fast absolute pose estimate of the robot.

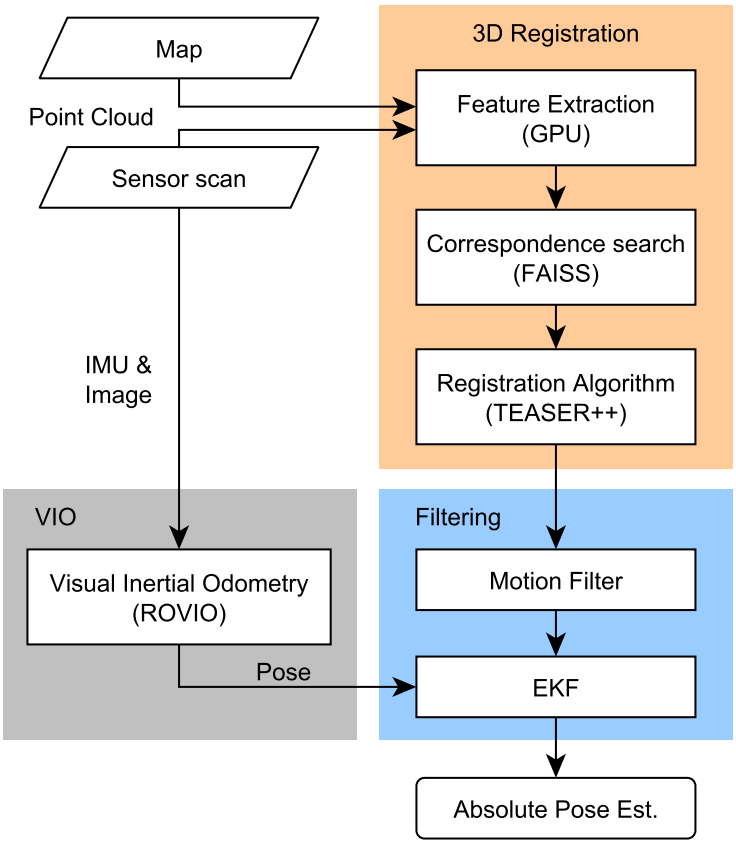


Figure D.1. The pipeline of the proposed absolute localization system. The main parts of the system are the Feature extraction (using either 3DSmoothNet or FCGF), robust 3D point registration (using TEASER++), and the Extended Kalman Filter which fuses the pose estimate from the registration algorithm with a relative visual-inertial pose.

The 3D registration method is executed in three consecutive steps; (1) Feature extraction, (2) correspondence search and (3) the registration algorithm. In the first step of the registration process, 3D features are extracted which describe the geometric structure at a given point. The feature descriptors for the map are only calculated once whereas the features from the sensor scan are calculated at each arrival of the scan.

D.2.1 Feature extraction

The investigated feature descriptors for this paper are Fast Point Feature Histograms (FPFH) [135], 3DSmoothNet [136] and Fully Convolutional Geometric Features(FCGF) [138]. The first method, FPFH, is an extension of Point Feature Histograms(PFH), and is a well known 3D point descriptor. It uses normal information of the neighboring points to generate a histogram that reflect the characteristics of the neighborhood around each point. SmoothNet is a deep learning-based descriptor and consists of two parts.

The system starts by calculating for every point a voxelized smoothed density value (SDV) representation, which limits the sparsity of the input voxel grid. To produce orientation invariance the SDV is aligned to a local reference frame (LRF), similarly to hand-crafted 3D descriptors. The resulting SDV is then used as an input to a siamese deep learning architecture working on stacked convolutional layers and the output unit length feature vector is produced by a batch normalization layer followed by an $l2$ normalization. Each input interest point is, therefore, assigned with a 32-dimensional feature vector. In this work, to make the 3DSmoothnet competitive with the much faster FCGF, we randomly select approximately 50% of the pre-existing map points and 25% of the live robot view points. We have verified that this way can both limit the computational cost and provide enough data to ensure correspondences.

FCGF is a 3D fully-convolutional network, without the need for prepossessing. FCGF is based on sparce tensors and sparse convolutions. It uses a UNet structure with skipped connections and residual blocks to extract sparse fully-convolutional features. In our paper FCGF is setup to its standard use of metric learning losses based on Hardest-Contrastive Losses. On the 3D match dataset [137], FCGF outperformed all other classical and machine-learning based feature descriptors —both in speed and feature match recall.

D.2.2 Correspondence search

Feature correspondences between the map and the sensor scan are then determined based on the GPU-based KNN algorithm *Facebook AI Similarity Search*(FAISS)[139]. Due to some inaccuracy's in the feature descriptors and the geometric similarities of the the environment, incorrect matches will arise, and are consider as noise in the correspondence set.

D.2.3 3D Registration

Due to the nature of the environment, i.e. similar structural areas, and imperfect feature descriptor, the produced correspondence points include substantial noise. Therefore, we should use an algorithm that allows for outlier rejection. We opted for the TEASER++ [140] which is a robust 3D registration algorithm, which is able to provide accurate transformation between two corresponding point sets —in our case robot's view and map points— even in the presence of high percentage of outliers.

The following brief description of the registration algorithm was also described in [3], but is included here to allow for a standalone paper.

In an ideal case, where no outliers exist among the correspondences, the registration can

be defined as a nonlinear least squares solution, as follows:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left\| \mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t \right\|^2 \tag{D.1}$$

, where the minimization is performed over the scale s, the rotation $\mathbf{R}$, and the translation t. Due to the metric environment of the use case, a scale of 1 can be assumed and, therefore, no additional computational resources will be spent on estimating the scale. The 3D correspondence points are notated as $\mathbf{a}$ and $\mathbf{b}$, where (a_i, b_i) represent a correspondence pair between the map and sensor point cloud features and N represents the total number of corresponding points. Due to the noisy measurements of the point cloud reconstruction, a Gaussian noise with isotropic covariance described by σ^2 is included. However, for most real world cases, correspondences with zero outliers cannot be safely assumed [140] and therefore the robust registration can be performed using a Truncated Least Squares function D.2, as follows:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \min \left(\frac{1}{\beta_i^2} \left\| \mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t \right\|^2, \bar{c}^2 \right) \tag{D.2}$$

Equation D.2 describes a least squares solution of measurements with small residuals ($\leq \bar{c}^2$), where β_i is a given bound of the noise. This noise bound is set as the maximum error expected from an inlier or 3σ. Large residuals ($> \bar{c}^2$) are discarded. In our experimental setup, $\bar{c}$ is set to 1. To simplify the solution, the rotation and translation are decoupled in TEASER++, as expressed by equations D.3 and D.4 respectively.

$$\hat{\mathbf{R}} = \arg\min_{\mathbf{R}\in SO(3)} \sum_{k=1}^{K} \min \left(\frac{\left\| \bar{\mathbf{b}}_k - \hat{s}\mathbf{R}\bar{\mathbf{a}}_k \right\|}{\delta_k^2}, \bar{c}^2 \right) \tag{D.3}$$

Firstly, to estimate $\hat{\mathbf{R}}$, we want to minimize the distance between corresponding points expressed as $\bar{\mathbf{b}}_k$ and $\bar{\mathbf{a}}_k$, with a bounded noise δ_k. Using the estimated rotation, the translation can then be determined by equation D.4. The translation is acquired component-wise, i.e., the entries t_1, t_2, t_3 of $\mathbf{t}$ are computed independently. Large residuals ($> \bar{c}^2$) are, as in the previous equations, discarded:

$$\hat{t}_j = \arg\min_{t_j} \sum_{i=1}^{N} \min \left(\frac{(t_j - \left[\mathbf{b}_i - \hat{s}\hat{\mathbf{R}}\mathbf{a}_i \right]_j)^2}{\beta_i^2}, \bar{c}^2 \right) \tag{D.4}$$

An analytic derivation of the aforementioned formulation can be found in [140]. The 3D registration method assumes no knowledge of prior poses and incorrect registrations will therefore greatly affect the pose estimate.

D.2.4 Filtering

To remove the pose outliers from incorrect registrations, a simple motion filter is introduced, which remove pose estimates with movements larger than a predefined threshold. The filtered poses from the 3D registration is then used in an Extended Kalman Filter along with the VIO pose. The VIO pose is estimated using ROVIO and is updated at a faster rate than the 3D registrations. The resulting pose estimates from the Extended Kalman Filter are therefore both fast and accurate.

D.3 Experimental Setup

The proposed localization system is tested offline on a data-set of a representative structured and confined space, specifically a mock-up model of a topside water ballast tank (WBT) from a marine vessel. Most often CAD drawings of these confined spaces are available, and a map can be extracted from the drawings. The environment, or the *map*, for the experiments are represented as the dark points in figure D.2. This mock-up model consists of two compartments with similar structural geometry to reflect real-world conditions and challenge the 3D localization system. The x,y,z size of the map is respectively 5.30, 2.9, 1.75 meters.

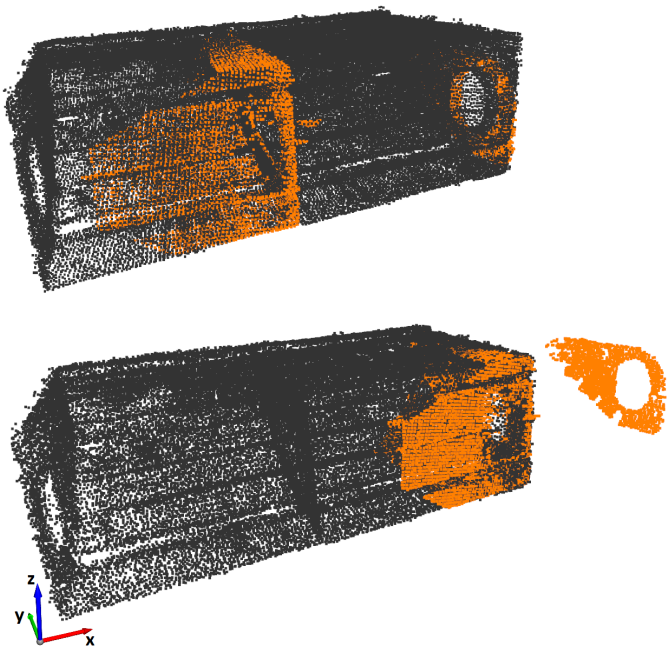


Figure D.2. Illustration of a correct(top) and incorrect(bottom) registration with the point clouds used in the experiments, where the map is shown in black and the scan points in orange. The structural similarity of the environment can cause errors when matching without prior location information. These incorrect registrations are in this paper avoided using simple motion filtering techniques.

The experiments are done on a data set collected with the Azure Kinect Time of flight camera inside the mockup model. The Kinect Camera generates RGB images and 3D point clouds at 15 Hz. To estimate the position of the camera, the point cloud arriving from the sensor is registered to the point cloud of the map. The registration is based on TEASER++ [140] and uses features generated by either FPFH, 3DSmoothNet [136] or FCGF where the last two methods utilizes the described deep learning architectures to extract 3D features from point clouds. Both deep learning methods were trained on the 3D Match dataset [137].

FPFH was tested using the Open3D [150] implementation with its standard 33 dimensional feature descriptor on down-sampled clouds using a voxel size of 0.04 meters, and a normal radius estimation of 0.08 meters. The size of the feature descriptors for 3DSmoothNet and FCGF were both 32 dimensions, and the point clouds used in the experiments were downsampled with a voxel size of 0.04 meters. FCGF was run with a kernel size of 3, and

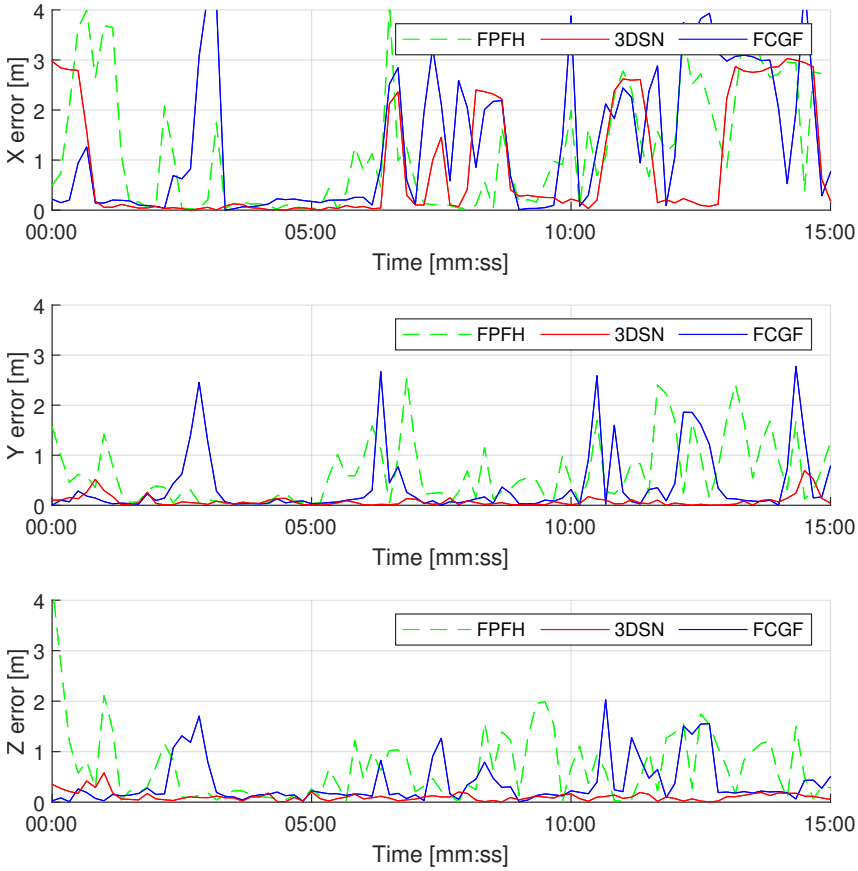


Figure D.3. The absolute position error for the 3D registration, using only Teaser++ and one of the three different feature descriptors; FPFH, 3DSmoothNet(3DSN) or FCGF

the algorithm inherently extract features from all available points in both the map (112.739 points) and sensor scan. 3DSmoothNet was run with a SDV voxel size of 16, and grid size of 0.8 meters, 55.000 random keypoints in the map, and 1000 randomly selected keypoints for each scan. The downsampling of the map points and the random selection of the robot points in the 3DSmoothnet was decided so that results would be adequate while keeping the computational time withing reasonable —in terms of online operation— bounds.

From the absolute position error in figure D.3 it can be seen that large sudden errors occur for all methods on especially the x-axis. This is mostly due to the geometric similarity of the environment, and an example of such an geometric similarity between a correct and incorrect match is shown in figure D.2. From the raw unfiltered 3D matching the best accuracy is obtained from 3DSmootNet, followed by FCGF and FPFH. The accuracy of 3DSmoothNet comes at a significant higher computational resource cost than FCGF as it can be seen from the compute time plot in figure D.5. According to the authors of [138], FCGF should however be more accurate than 3DSmoothNet when applied to the 3D match test dataset. The results shown here could indicate that 3DSmoothNet might be more generalizable than FCGF, when applied to datasets which the networks have not trained on.

The outliers from the 3D registration can to some degree be filtered by the motion filter, depicted in figure D.1, which rejects updates with a velocity larger than 0.3 m/s. The motion filtered 3D registrations is fused with an VIO pose estimate in an Extended Kalman

filter using a ROS implementation of a generic Extended Kalman Filter [151]. The result of the whole localization pipeline is shown in figure D.4, where it is clear that the filtered EKF pose error is less than the unfiltered pose estimate in figure D.3. The mean and standard deviation of the tested methods can be seen in table D.1, where 3DSN+T is only using 3DSmoothNet features and Teaser++ for 3D registration, without any filtering, as shown in figure D.3. The mean and standard deviation of the full localization pipeline is depicted as 3DSN+EKF or FCGF+EKF depending on which feature descriptor is used on the point clouds.

Table D.1. Euclidean positioning error

	Mean (m)	Std.(m)
FPFH	1.8312	1.3189
3DSN+T	1.0549	1.1782
FCGF+T	1.5928	1.4171
ROVIO	0.1724	0.0566
3DSN+EKF	0.0999	0.0497
FCGF+EKF	0.1211	0.0617

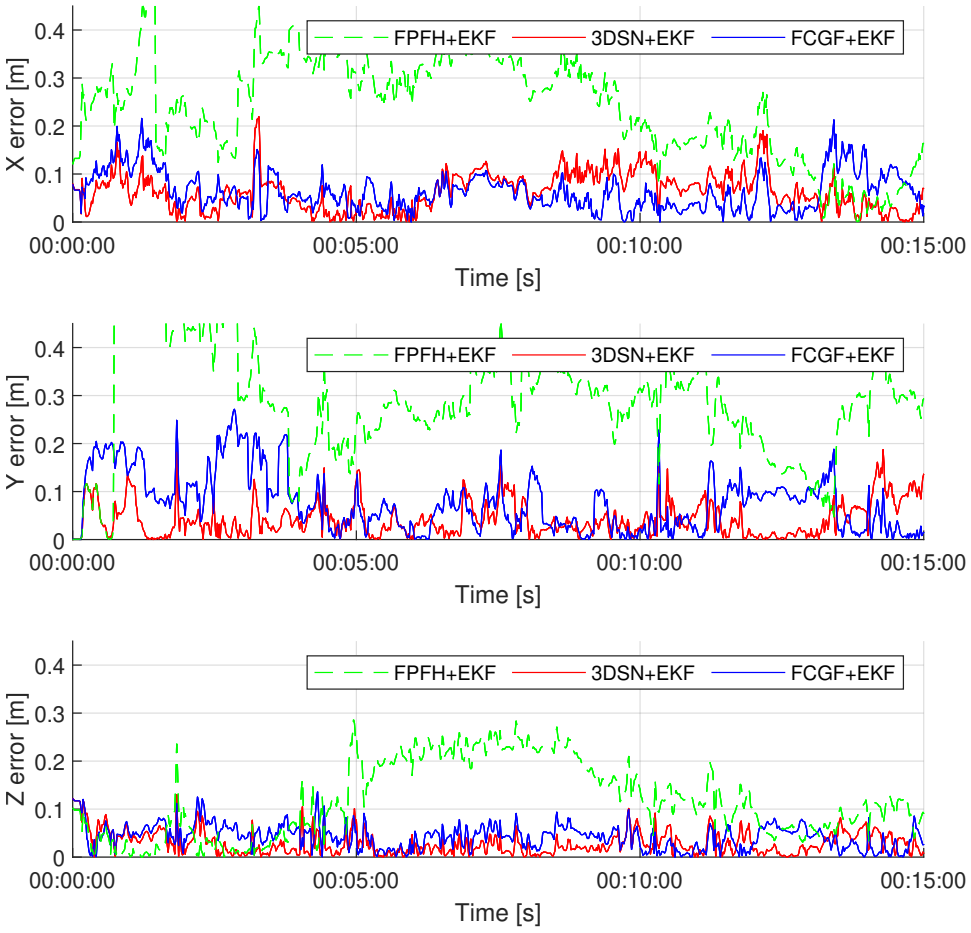


Figure D.4. The absolute position error for our localization system, utilizing the full pipeline with motion and Extended Kalman filter, using one of the three different feature descriptors; FPFH, 3DSmoothNet(3DSN) or FCGF.

D.4 Computational time

An important aspect to localization systems is the required computational resources and time to acquire a pose estimate. The average computational time for the localization pipeline using the three different feature extraction methods can be seen in figure D.5.

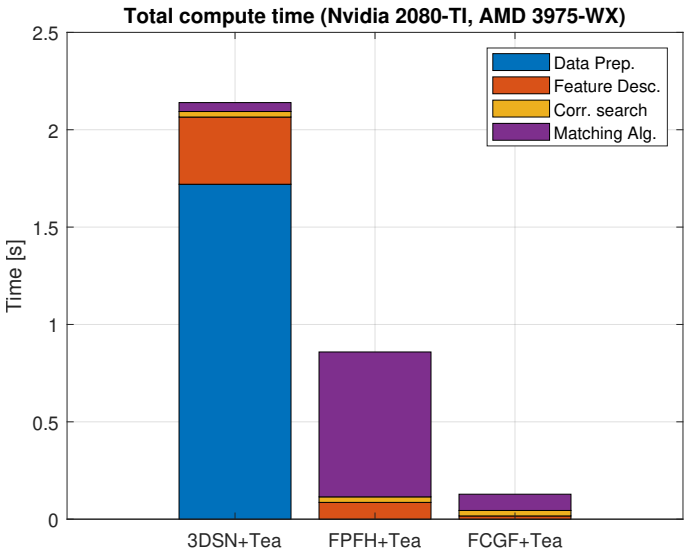


Figure D.5. The average computational time for each element in the 3D registration methods; (i) TEASER++ registration using 3DSmoothNet features (3DSN+Tea), (ii) TEASER++ registration using FPFH features (FPFH+Tea), (iii) TEASER++ using FCGF features (FCGF+Tea).

The correspondence search and feature descriptor calculations were all conducted on a Nvidia 2080TI GPU, and the Teaser++ registration and 3DSmoothNet data preparation (SDV+LRF) calculations were done on an AMD 3975-WX CPU. On the current system, interchanging 3DSmoothNet with FCGF reduces the computation times with a factor of 6.7, with only minor increase in the filtered localization error, as depicted (as FCGF+EKF) in table D.1.

D.5 Conclusion

In this paper we proposed an absolute localization system utilizing deep geometric feature descriptors and VIO relative poses, for localization within confined spaces. Two different methods were tested —using the same training data from 3DMatch—, where 3DSmoothNet performed the best in terms of raw 3D point accuracy and FCGF was by far the fastest while still performing adequately. The experiments indicate that using deep geometric features instead of Fast Point Feature Histograms, the absolute localization performance in both accuracy and speed could be increased after simple filtering. Future improvements to the system can be achieved by training the networks to the specific operating environment, since the results indicate that the fastest performing network was sensitive to the used traning set.

Chapter 5

Uncertainty Aware Absolute Localization

Up to this chapter, the uncertainty of the 3D registration has not been incorporated. In the literature, accurate estimation of the uncertainty of 3D point cloud registrations has been investigated [156, 157, 158], with a focus on ICP. Many of these methods have been computationally intensive and have a tendency to underestimate the actual uncertainty [159]. The novel deterministic method, Stein-ICP[159], can estimate the uncertainty of 6-dof point-cloud registrations utilizing the parallelizing capabilities of GPU.

Due to the lack of uncertainty estimation in my previous 3D registration methods, Stein-ICP is therefore introduced in the 3D registration pipeline in **article E**. The addition to the 3D registration pipeline is shown as the uncertainty estimation block in Figure 5.1.

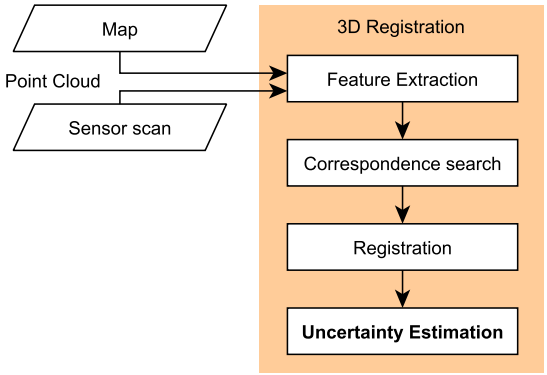
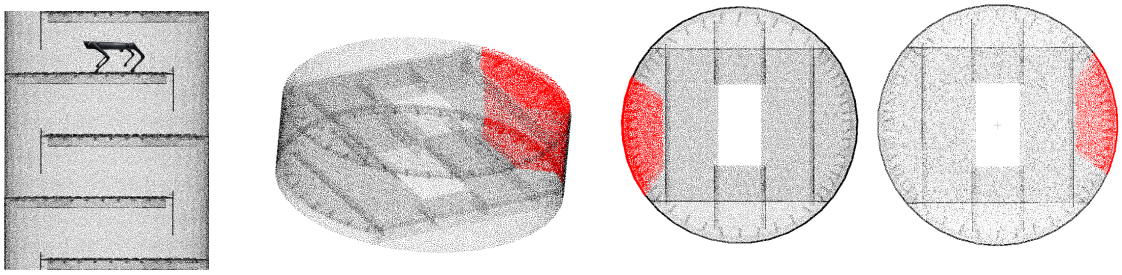


Figure 5.1. The updated version of the 3D registration pipeline shown in Figure 4.2, where the uncertainty module has been added to include the uncertainty in the resulting absolute pose estimate.

Article E also includes a waypoint-based inspection system, where artificial defect points are detected and inspection data are collected of each defect. The entire autonomous inspection system is tested onboard a small UAV within the mockup model of a water ballast tank. The results demonstrate a system able to localize and traverse the confined space, where corrections to a drifting VIO are correctly applied. In addition, the system locates defects within the confined space within some varying noise-bound due to a poor timing issue from the point cloud sensor.



(a) Cross section *(b)* Ambiguity registration example of a single column section

Figure 5.2. Example of ambiguity of a typical distillation column. The grey points represent the map and the red points a point cloud from the sensor. The red point cloud can reasonably be registered to multiple locations in the map, from which the ambiguity problem therefore arises.

For environments with an even higher level of similarity and ambiguity than ballast tanks, the Stein-ICPs registration approach further adds value in its approach to determine multiple successful registrations. Figure 5.2b illustrates an example of such a scan (red) within a highly symmetrical section of a simulated chemical distillation column. For the human reader, it is clear that the scan has the potential to be registered to multiple locations in the gray point cloud map. However, using the previously discussed methods based on feature descriptors, the system will suffer from incorrect data association in the correspondence list and therefore return severely incorrect registrations, which will furthermore impact the pose estimate of the robot. Therefore, a different approach is needed that can handle this kind of ambiguity. Stein-ICP can provide a good solution to the example presented in Figure 5.2, where the focus, for the sake of simplicity, will be limited to only 1D in which we try to register the red point cloud by rotating around the center point of the circle. Stein-ICP will perform the registration for each of its particles, as described in **article F**, resulting in the Kernel Density Estimate (KDE) plot shown in Figure 5.3. The plot illustrates that two solutions are the most prominent with a rotation of around -80 and 100 degrees, which fit our intuition from looking at Figure 5.2b. The Gaussian is also plotted for the sake of completeness, and it is clear that this is not an optimal representation of the uncertainty.

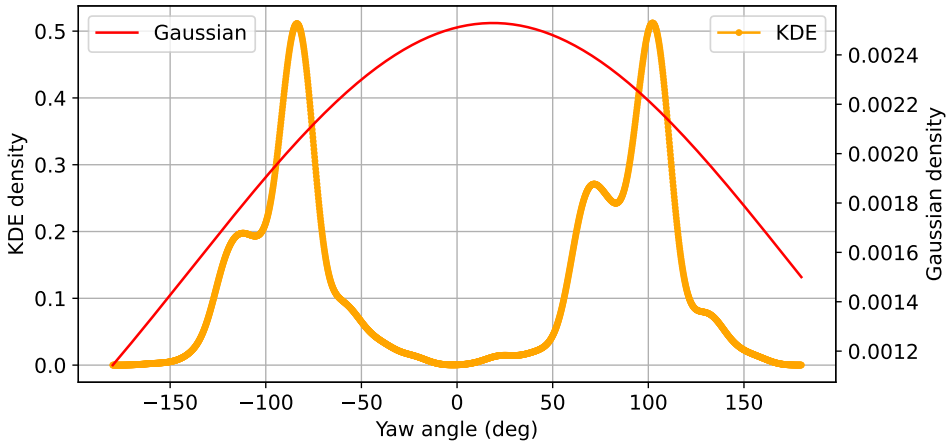


Figure 5.3. Illustration of the Gaussian and the KDE for the yaw axis of the point cloud registration shown in Figure 5.2b. Two equally suitable registration solutions arise at -80 and 100 degrees of yaw rotation, which illustrates the problem of ambiguity.

In **article F** a localization system was designed for these kind of highly similar and feature-few environments, using a modified version of the Stein-ICP approach as one of its primary contributions. The goal of the article is to provide an absolute pose estimate while a legged robot traverses the confined space of a distillation column. In the article, the ambiguity is mitigated to some extent by detecting a so-called downcomer inside the structure, which can be seen as the vertical plate in front of the robot in figure 5.2a. Registrations are only attempted when at least part of this unique feature is in view. However, due to structural elements, such as cross-sectional beams and cylindrical walls, correctly associating classic feature descriptors is still challenged by the high level of similarity in the environment. Stein-ICP is therefore chosen as the approach for the 3D registrations in this article, and Stein-ICPs 6 degrees of freedom are optimized to 3, to decrease the computational requirements. Stein-ICP is furthermore improved to use the previously mentioned KDE to provide the most optimal pose estimate to an extended Kalman filter in the localization pipeline. The Kalman filter fuses this pose estimate with a velocity estimate from a VIO system to provide a fast and accurate absolute pose estimate. The results from both simulated tests and real-world tests indicate a robust accuracy of the pose estimate with few outliers in the feature-few environment.

Article E

Autonomous GPU-Based UAV for Inspection of Confined Spaces: Application to marine vessel classification

Rune Y. Brogaard[1], Evangelos Boukas[2]

[1]FORCE Technology, Park Allé 345, Brøndby, Denmark

[2]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

Abstract:

Inspection of confined spaces poses a series of health risks to human surveyors, and therefore a need for robotic solutions arises. In this paper, we design and demonstrate a real-time system for collecting 3D structural and visual data from a series of inspection points within a prior map of a confined space. The system consists of a GPU accelerated 3D point cloud registration and a visual inertial odometry estimate fused in an Unscented Kalman Filter. Using the state-of-the-art deep learning-based feature descriptors, FCGF – and the robust Teaser++ 3D registration algorithm – point clouds from a narrow field of view, time-of-flight, camera can be registered to a prior map of the environment. The uncertainty of the system is furthermore estimated on the basis of the novel GPU-based Stein ICP algorithm. Visual defects, represented by augmented reality fiducial markers, are automatically detected during inspection, and their positions are estimated in the map frame of the confined space. The performance of the system has been evaluated in real time onboard a small UAV, within a mock-up model of a water ballast tank from a marine vessel, where the UAV was able to navigate and inspect the ambiguous and featureless environment.

E.1 Introduction

Inspection of dark and confined spaces can be a challenging task for both human and robotic applications. Inspections of these spaces are performed mainly by humans due to the often complex structures and limitations of current robotic capabilities. The environment

can often be considered dangerous due to the variability in atmospheric conditions, the risk of falling, and the often unknown integrity of the structure. Similarly, this line of work can also be tiresome and dirty, further increasing the risk of human error, such as missing critical defects. Therefore, it is important to minimize human involvement during inspection of confined and structurally complex spaces, to eliminate human risk, and to maximize repeatability of inspections. The repeatability of an automated inspection solution can also allow analysis of the trend of defects in critical areas.

For the inspection of the water ballast tanks, three modes of operation are of interest, namely 1) Mapping and exploration of the environment, 2) Visual close-up inspections, and 3) Contact-based thickness measurement of the metal structure. The authors of [7] have shown how water ballast tanks can be mapped using an exploration-based approach where unknown spaces are used as an element in the RRT cost function. Therefore, the focus of this paper is on autonomously performing visual inspections in an already known environment. The environment can be known from either the CAD drawings of the ship or from the construction of a map as shown in [7].

The goal of this paper is to autonomously fly an inspection mission in a water ballast tank, while collecting inspection data in the process. The proposed inspection system to solve this problem can be seen in Figure E.5, where this paper covers the entire pipeline except the data evaluation part. In-depth research on deep learning-based detection and evaluation of images collected from the UAV is covered by [77, 91]. Most autonomous systems require a good localization module, and this will therefore be the backbone of our system. A large part of our proposed inspection system is dedicated to both localizing the UAV itself, but also localizing detected defects. Localizing within structured, confined, and ambiguous spaces poses a range of challenges. These spaces often contain low lighting conditions and poor visual features, therefore limiting visual tracking or recognition. Likewise, the nature of industrial man-made spaces is often built with a certain repeatable pattern in mind, challenging classical 3D geometric pattern recognition and subsequent data association. The authors of [4] show that the well-known Fast Point Feature Histograms (FPFH) performed poorly in these structured environments, compared to deep learning-based geometric feature descriptors. The inspection pipeline handles a range of these challenges by utilizing prior information from the CAD drawing of the industrial space, which in this case is the water ballast tanks of marine vessels. The pipeline, furthermore, uses a Kalman fused combination of Visual Inertial Odometry and 3D deep learning-based geometric features, collected, respectively, from an Intel T265 stereo camera and L515 ToF camera.

It is important to note that the designed inspection system has the potential to be utilized on various kind of robotic platforms, where in this paper we have tested the pipeline on a small Unmanned Aerial Vehicle (UAV). Therefore, the purpose of the UAV is to visit planed inspection points inside the ballast tank and collect image data of defects. In this paper, the defects are marked using wall-mounted ArTags, at typical positions of structural stress concentration.

Part of the localization system has been tested offline in [4], whereas this paper focus on real time execution of a whole automatic inspection pipeline. Due to less requirements for realtime performance, the automatic evaluation of the inspection data can still be obtained offline using different methods based on machine learning such as described in [77, 91].

The main contributions of this work are as follows:

1. The real-life online application of GPU and deep learning based localization using 3D geometric information, allowing the autonomous operation of inspection robots in featureless environments.
2. Proving that a faster but inferior deep geometric feature descriptor is sufficient for the accurate online localization of aerial inspection robots under the condition that the system includes a robust registration algorithm capable of providing accurate transformations even in the presence of high percentage of outliers.
3. The first-time application of a well-calibrated (3D point cloud matching) uncertainty estimation algorithm running on a GPU in an online robot localization case.

E.1.1 Related Work

Entering confined spaces is considered a high-risk operation for humans, and therefore the need to use robots has emerged. The most common approaches are submerged Remotely Operated Vehicles (ROV), Magnetic Crawlers, or Unmanned Wheeled Robots, and to some extent, Aerial Systems.

Research on ROV applications, such as [36, 160, 161], has focused mainly on inspecting the exterior of the hull. This has been done by submerging the vehicle near the vessel and either manually controlling the ROV or by using wheel encoder data and to some degree visual odometry. For the confined spaces in the ballast tanks, ROVs are, however, not considered as a viable solution. The ballast tanks are rarely filled with water, and visibility of the water in the tanks is often limited due to impurities in the water such as mud and marine growth.

The authors in [103, 94] have investigated the usage of crawlers and small legged robots for inspection of marine vessels. The authors of [103] have designed a magnetic track wheel robot, which is capable of autonomously building a mosaic image of a vertical inspection run on a planar surface of the cargo bay. This design is limited by a simple odometry localization system, but also by its ability to traverse obstructions in the ballast tanks such as ladders, beams, and bulkheads. Instead, a small legged robot was designed in [94], which uses electromagnets on its feet, and is able to traverse tight areas and obstacles, as long as they are magnetic. To our knowledge, they do, however, not provide a localization or navigation system for autonomous control of the robot. The authors of [42] build on a similar magnetic approach as [103], but uses an external pose estimate for reference, which is not considered possible for the complex and confined space of the water ballast tanks.

The challenges of traversing confined spaces using aerial systems have been described in [162, 77], where they mention reliable absolute localization and lack of remote control as some of the major difficulties. It is therefore clear that some level of autonomy is required for the UAV. The authors of [16, 14] solve the localization issue by implementing Ultra Wide Band(UWB) satellites inside the confined space to give them an absolute global coordinate system for the UAV. This proves to provide accurate results, but installing UWB electronic hardware in the water ballast tanks is in our case not considered a viable solution. Other aerial systems such as [163, 164] use cameras and touch-based sensors to demonstrate the ability to traverse small man-way-sized ventilation shafts, but

have not focused on collecting inspection data during their flight. The authors of [165] provide a path planning solution for autonomous inspection of the outside of marine hulls, using multiple UAVs to first perform a general inspection and secondly a close-up inspection of detected defects. The system was not tested on real vessels or in more constricted environments such as confined spaces but, rather, in simulated environments. Other research [166] has focused on exploration and mapping of the environment using a combination of Simultaneous Localization and Mapping(SLAM) and Convolutional Neural Networks (CNN). This research was conducted using a similar sized UAV as ours, but was however only tested in a well lit office environment.

The authors of [56] used LIDARs, cameras, and IMU data from the onboard AscTec flight controller to navigate around the cargo hull of a large marine vessel. The UAV used a 2D lidar to estimate the distance to the surrounding walls and a 1D lidar to estimate the altitude inside the cargo hull. The system was tested within both the cargo hull and the top-side ballast tanks, but a user was still required to give high-level commands to the UAV through a base station using a joystick. The system was further expanded in [87] to use multi-threaded Binary descriptor-based Image MOSaicing (BIMOS) to create a single overview image of the inspection surface, using ORV features and Keyframes.

The research carried out in [167] implements an autonomous Unmanned Aerial Fire Detection System, for marine vessels. It can navigate inside the main areas of the ship, such as corridors and doorways, by utilizing a prior map of the environment. A particle-filter-based localization algorithm was used to localize the UAV in a generated 3D octomap. They are able to localize the fire using a thermal camera, but point to some failures and limitations due to drifting odometry and poor 3D registrations between the sensor-scan and the prior map.

In [116] the authors used a UAV to explore an underground mine environment, to locate objects of interest, which in their case were humans in need of rescue. The solution is based on the fusion of LiDAR data with thermal vision frames and inertial measurements. The system proved capable of autonomously navigating the mine environment, however, the repeatable structure and ambiguity of the ballast tanks do not allow for a simple transfer of the system to new domains.

E.2 System description

E.2.1 Unmanned Aerial Vehicle

To test the entire inspection navigation pipeline, an Unmanned Aerial Vehicle (UAV) is custom built to be able to enter through the man-way in the water ballast tanks. The UAV shown in figure E.1 is based on a Lynxmotion Crazy2Fly Drone Kit equipped with a Pixhawk 4 Mini Flight Controller Unit(FCU), an Aeeon PICO-WHU4 I5 PC, and a Nvidia Jetson Xavier NX that functions as the GPU of the system. A protection cage is installed around the UAV to protect the propellers from potential impacts with the surrounding environment. The cage of the UAV increases the footprint of the UAV to 500 x 500 mm, with a height of 140 mm.

The UAV is also equipped with an Intel L515 Time-of-Flight camera and two Intel T265

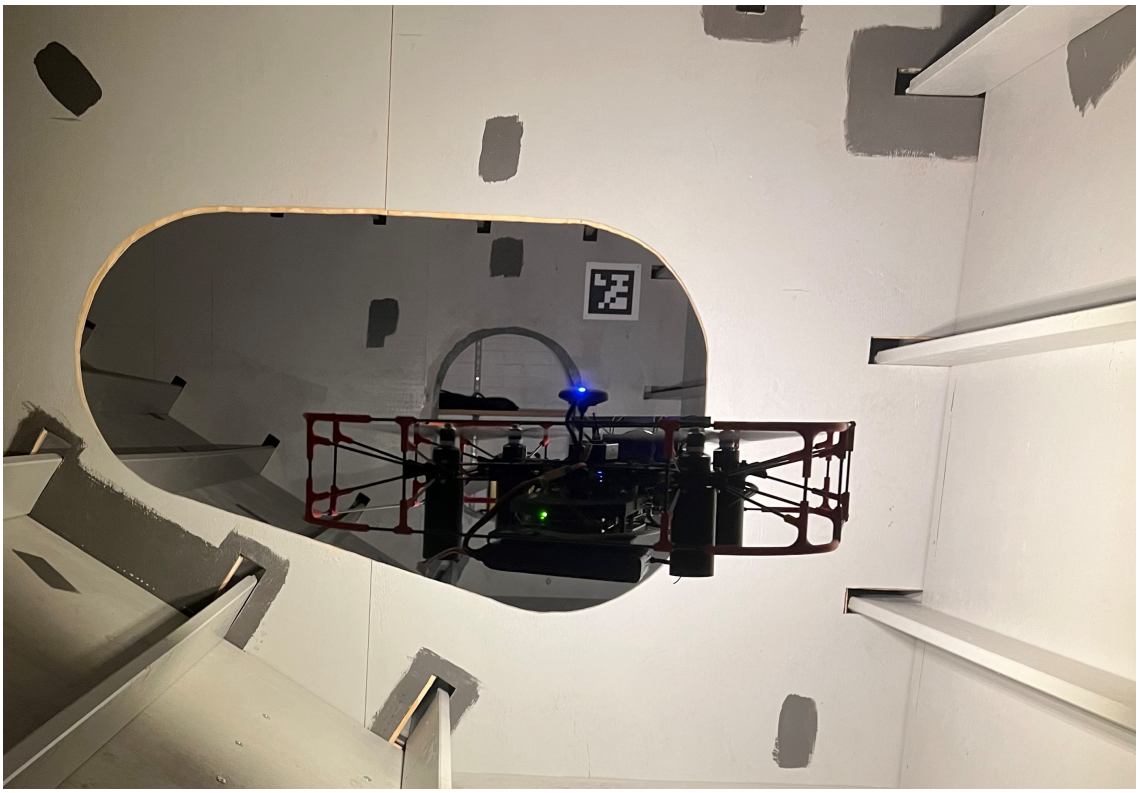


Figure E.1. The Unmanned Aerial Vechicle used in the experiments of the autonomous inspection system. A simulated defect is depicted as an AR tag in the next ballast tank compartment.

VIO cameras. The front facing T265 is acting as the VIO input to the localization system, where as the down facing camera is completely decoupled from the localization system, to only provide the ground truth estimate from a series of AR tags installed on the bottom of the tank. The UAV uses the two computers onboard to process sensor data and communicate with onboard Flight Controller Unit (FCU). The main computer is the AAEON PICO-WHU4, and the second processing unit is the Nvidia Jetson NX development board. The onboard PICO PC handles the communication to the FCU and drives the data collection of the T265 and L515 cameras. The L515 point cloud is down-sampled using a voxel grid of 0.05 m, and the point cloud is then transmitted to the Nvidia Jetson over a cabled Ethernet connection. The Nvidia Jetson handles the majority of the localization pipeline by utilizing the Graphical Processing Unit. The latter system also runs the 3D registration part of the Localization pipeline, where especially the correspondence search and feature descriptors benefit from the parallel computation power. The low-level control of the UAV is done by the FCU Pixhawk 4 Mini[168], running the PX4 1.10 software stack, where the communication to the AAEON PC is done by a wired RS232 connection. The PC sends 20 Hz pose estimates and waypoints to the FCU, which then ensures the robot is on the commanded position with an acceptance threshold set to +/- 10cm. The PICO and the Jetson run Ubuntu 18.04 with ROS Melodic[169], where the PICO PC is running as the ROS master and the Nvidia Jetson as a secondary device, with a wired Ethernet connection. The defects, represented as April Tags, are detected using the ROS package AprilTag 3 visual fiducial detector[170] on the PICO PC using the RGB image stream from the L515. Using the camera matrix, the 2D April tag is transformed into the 3D position of the camera frame and then transformed into the map frame using

the absolute pose estimate from the localization pipeline. It is important to note that no April tags are used for the actual localization part of our system.

E.2.2 Environment

The proposed system has the potential to operate in a wide range of confined spaces, but the focus of this paper has been to verify the system in a water ballast tank of a marine vessel. To enable rapid development and testing –and due to temporary access restrictions to vessels– a mock-up model of a topside ballast tank has been used for the experiments. A map of the ballast tank is represented as the gray point cloud shown in Figure E.2. The ballast tank consists of two compartments with a bulkhead/webframe as the divider. A registered scan from the L515 is shown as the orange point cloud in the figure.

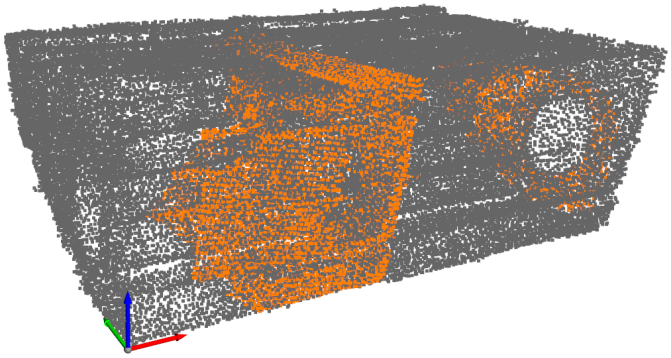


Figure E.2. Illustration of the mock-up model of the water ballast tank, shown as the grey point cloud. An example of an aligned sensor scan is depicted with orange points.

To provide a reasonable ground truth estimate, a series of April Tags are mounted on the floor of the ballast tank, as shown in figure E.3. The positions of these tags were first manually mapped and then used in the localization part of the TagSLAM package[171], to provide the ground truth pose of the UAV.

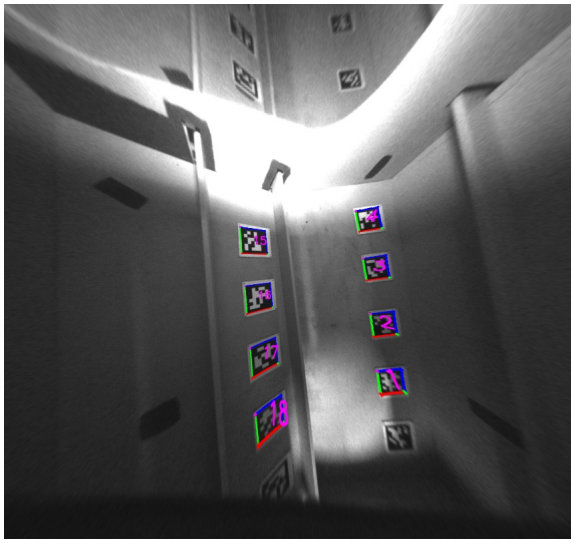


Figure E.3. Undistorted and image from the T265 camera, with the floor mounted ground truth tags visible. The tags are only used for the ground truth estimation and not for the actual localization pipeline.

The defects the UAV needs to detect and localize are represented by 6 April Tags mounted on the walls within the tank, and in the field of view of the camera at the specified inspection points, as shown in figure E.4. The inspection points are defined on the basis of known areas of the ship that must be inspected according to shipping regulations. These inspection points are often areas that are prone to deterioration or failure, due to the high material stresses within the structure of the ship.

Figure E.4. Example of an April Tag defect detected by the Camera

E.3 Inspection pipeline

To autonomously perform the inspection, the system is designed as shown in Figure E.5. During the UAV startup, a list of inspection waypoints is loaded onto the onboard PC and a point cloud map of the environment, with its extracted geometric features, is loaded into the GPU memory of the Nvidia Jetson.

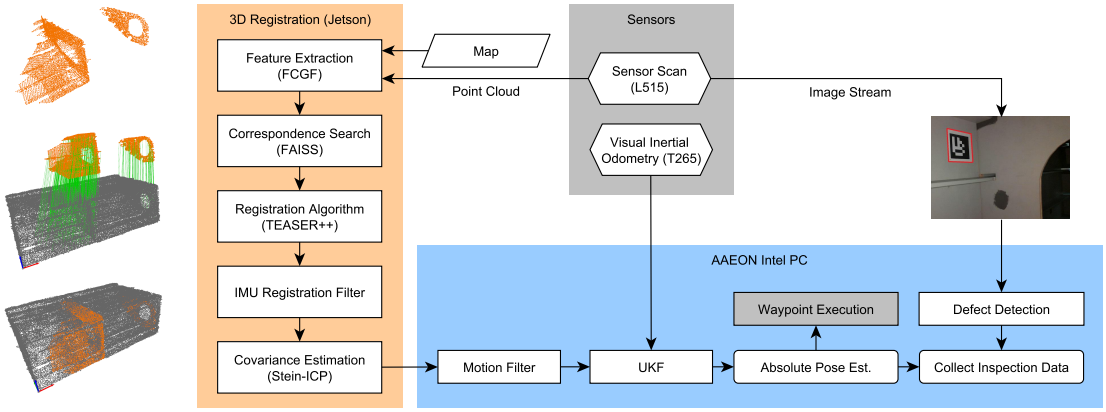


Figure E.5. Overview of the inspection system running onboard the UAV. 3D point cloud registrations between a known map and scan from the L515 sensor are fused with a VIO velocity estimate in an UKF to give an absolute pose estimate in the map frame. A defect detection system is continuously evaluating image data from the sensor to provide an estimate of defects within the map.

E.3.1 Inspection waypoint execution

The map of the environment is provided based on the CAD drawings of the ship and an inspection path can be generated for the UAV based on inspection points. Inspection points are defined in the specifications of the classification societies. The generated route

is, thereafter, executed by the UAV using the Pixhawk FCU using the localization pipeline of section E.3.2 as its absolute pose estimate. During inspection execution, the UAV will arrive at a series of inspection points, where along its path it will look for defects and collect visual data in the form of RGB images and surface point clouds for any detected defects. The RGB images and surface point clouds will later serve as the evaluation and decision base for onshore inspection personnel. Although not included in this paper, for the sake of conciseness, we have in the past automated the evaluation of corrosion or defects using different machine learning methods as investigated by [77, 91].

E.3.2 Localization pipeline

A major part of an autonomous inspection system is its localization pipeline, as illustrated in Figure E.6. The pipeline can be divided in three sections; a 3D registration module, Visual Inertial Odometry (VIO), and lastly a filtering step. The 3D registration module generates an absolute pose estimate by registering the sensor scan (point cloud) to the map of the environment. The VIO provides a relative odometry estimate, where in our case the velocity is used as a state update in our filtering step. The filtering module of the pipeline first removes poor absolute pose estimates from the 3D registration based on a motion filtering threshold, and then fuses the inlier pose estimates with the velocity estimate from the VIO, in an Uncented Kalman Filter (UKF).

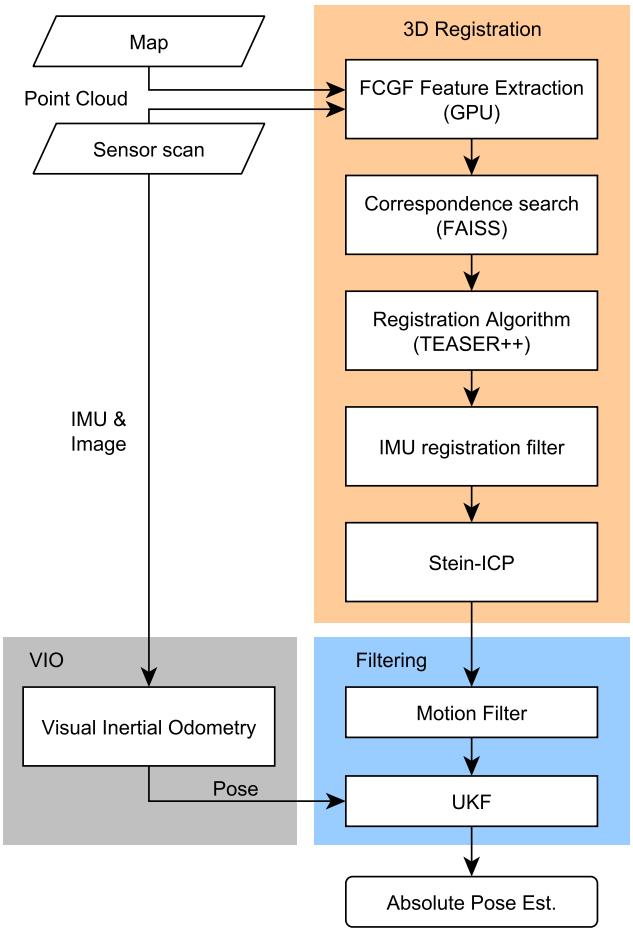


Figure E.6. Localization pipeline

Localization based on 3D registration

Due to the fact that the intended environments (i.e. ballast tanks of marine vessels) for this system are visually featureless, the localization pipeline in this work relies on 3D registration of —live acquired— 3D scans with pre-existing 3D models. While pair-wise registration is usually employed for 3D point cloud registration tasks [172], the algorithms proposed in the literature, mainly variants of ICP, are not suitable for implementation on an inspection UAV, due to the ICP's requirement of approximate prior knowledge of the model/target pose, the problem of local minima and due to its inability of providing uncertainty estimates (see sec.E.3.2) which is a must have for appropriate state estimation. Therefore, we follow the approach of feature-based 3D registration and specifically we employ deep learning feature extraction. Then we use a robust registration system, which is designed to accurately provide correspondences even in the presence of significantly more outliers than inliers. Finally, since the output of the 3D registration is used for state estimation, we integrated a state-of-the-art method for uncertainty estimation of 3D pointcloud registrations, allowing us to overcome a common problem of the 3D registration, i.e. uncertainty underestimation [159].

Feature extraction and data association The first step in the 3D registration module of the localization pipeline is to extract geometric features both in the map and in the sensor scan. Hand-crafted features have been very successful in providing some specific qualities, e.g. rotational invariance. A typical example is the FPFH[135] algorithm (and its predecessor PFH) which use a histogram of surrounding 3D points to calculate correspondences invariant to changes in orientation. We have tested such approaches in [4] and it was concluded that they do not perform well and are therefore not included in this investigation.

To overcome the trade-off between these different handcrafted features, we can employ learned features which, despite being less explainable, incorporate multiple qualities present in several custom designed features. Two such approaches are presented in this work, SmoothNet[136] which can achieve high accuracy but is computationally expensive — therefore used as a baseline— and FCGF [138] which is fit for online execution, albeit with less accuracy. We will briefly present both of the approaches for completeness.

SmoothNet is a deep learning-based descriptor with a focus on robustness as well as rotational and isometry invariance. Rather than implementing an end to end trainable network it has two distinct parts; a) the "preprocessing" of a pointcloud using Local Reference frames (LRF) and smoothed density value (SDV) representation and b) the fully convolutional part, which encodes a descriptor. To achieve rotational invariance, the neighbourhoods of randomly selected points in the pointclouds are structured in SDVs with LRFs defined by the neighborhoods themselves. Finally, the compact representation (3D SDV voxel grid) is fed into a siamese network to produce a descriptor.

In terms of the specific implementation of Smoothnet as a baseline for our system, the interest points were selected randomly —approximately 50% of all points in the map and 25% of the sensor point cloud. By randomly selecting points in both clouds, we limit the computational load while maintaining a high probability of acquiring correspondences.

As proven in our previous work on pre-recorded datasets [3, 4], Smoothnet is able to provide superior results in our application domain (WBT of large vesssels) which however comes at a significant computational cost. The cost is such that the approach could not be used for the online localization of our robot; a fully autonomous, self sufficient, power and time limited inspection robot in real life. While there is a fair argument to be made about the usage of Smoothnet on larger robots (e.g. a tethered aerial robot), we found that the method used in our robot, i.e. FCGF, is an order of magnitude more computationally efficient and that with the appropriate outlier rejection and robust estimation the system can provide adequate results.

The computational efficiency of FCGF stems from its structure, which is a one-pass 3D fully convolutional network, relying on sparse tensors and sparse convolutions. Additionally, FCGF does not rely on low-level pre-procecing of the input point clouds. Technically, the FCGF is a ResUNet (or Deep Residual UNET) architecture that uses skiped connections and residual blocks to extract fully convolutional descriptor features. By replacing the bridge of the UNET with residual blocks, the network is minimal in terms of parameters.

The most important contribution of the FCGF is the usage of "hardest-contrastive" losses. Contrastive loss is defined as follows: Similar features —or positive— should be as close as possible in the output feature dimension, and dissimilar features —or negative— must be at least a margin away. Instead of accumulating pairs of features (either positive or negative), the idea of hardest contrastive loss is to structure quadruplets using a positive pair with their "hardest" (or closer) negatives. This procedure is called hard negative mining. In the case of triplets (positive-positive-negative) rather than pairs, "hardest-triplet" losses can be structured. In our paper the network is setup to use standard metric learning losses based on "hardest-contrastive" losses.

The extracted features (whether Smoothnet or FCGF) are then searched for correspondences to features in the map using the GPU based KNN algorithm *Facebook AI Similarity Search*(FAISS)[139].

Registration

The extracted and associated geometric features from the scan and the map, are prone to a notable number of outliers due to the ambiguity of the environment. We, therefore, employ the robust 3D registration algorithm TEASER++[140, 141], which allow for outlier rejection. TEASER++ is capable of providing an accurate transformation between two corresponding point sets, even in the presence of a high percentage of outliers. The following brief description of the registration algorithm was also described in [4].

For ideal cases, where the 3D-point correspondence list contains zero outliers, the 3D registration can be defined as the following nonlinear least square solution:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left\| \mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t \right\|^2 \tag{E.1}$$

, where minimization is performed over the scale s, the rotation $\mathbf{R}$, and the translation t.

For our case with the ballast tanks, we can assume a metric environment with no significant scale changes between the scan and the map, and the scale factor s can therefore be fixed

to 1. The total number of correspondence points are denoted as N, and **a** and **b** represent two vectors of the corresponding pairs between the map and the sensor point cloud.

To include robustness of measurement noise in point clouds, a Gaussian noise with isotropic covariance described by σ^2 is included. So far we have assumed correct corresponding points, however, for most real-world cases, correspondences with zero outliers cannot be safely assumed [140] and therefore robust registration can be performed using a Truncated Least Squares function as stated by equation E.2:

$$\min_{s>0,\mathbf{R}\in SO(3),t\in\mathbb{R}^3} \sum_{i=1}^{N} \min\left(\frac{1}{\beta_i^2}\left\|\mathbf{b}_i - s\mathbf{R}\mathbf{a}_i - t\right\|^2, \bar{c}^2\right) \tag{E.2}$$

Equation E.2 provides a least squares solution to measurements with minimal residuals (no greater than $\bar{c}^2$), where β_i represents a given limit for the noise. This noise limit is either set as the maximum error allowed for an inlier or 3 standard deviations. Any measurements with large residuals (more than $\bar{c}^2$) are disregarded. In the experimental setup, $\bar{c}$ is set to 1. In order to simplify the solution, TEASER++ separates rotation and translation, as shown by equations E.3 and E.4 respectively.

$$\hat{\mathbf{R}} = \arg\min_{\mathbf{R}\in SO(3)} \sum_{k=1}^{K} \min\left(\frac{\left\|\bar{\mathbf{b}}_k - \hat{s}\mathbf{R}\bar{\mathbf{a}}_k\right\|}{\delta_k^2}, \bar{c}^2\right) \tag{E.3}$$

To determine the estimated rotation $\hat{\mathbf{R}}$, the first step is to minimize the distance between the corresponding points, expressed as $\bar{\mathbf{b}}_k$ and $\bar{\mathbf{a}}_k$, with a defined noise limit of δ_k. Once the rotation is estimated, the translation can be calculated using equation E.4. The translation is obtained on a component-wise basis, meaning the entries t_1, t_2, t_3 of $\mathbf{t}$ are calculated individually. Similar to the previous equations, any measurements with large residuals (more than $\bar{c}^2$) are discarded.

$$\hat{t}_j = \arg\min_{t_j} \sum_{i=1}^{N} \min\left(\frac{(t_j - \left[\mathbf{b}_i - \hat{s}\hat{\mathbf{R}}\mathbf{a}_i\right]_j)^2}{\beta_i^2}, \bar{c}^2\right) \tag{E.4}$$

An analytic derivation of the aforementioned formulation can be found in [140].

The transformation from the global TEASER registration is then compared with the roll and pitch attitude of the UAV. If the registration is offset by a given threshold, the registration is rejected, and the 3D registration pipeline is rerun with a new point-cloud scan from the sensor. If the registration is within the limits of the roll and pitch of the UAV, the uncertainty of the registration can then be calculated, along with a refinement step of the registration.

Uncertainty estimation To estimate the uncertainty, different methods are available with different properties. The uncertainty of the final 3D registration can be estimated using efficient closed form solutions, such as CELLO [156] and [157], which provides a

computationally fast solution, but sensor noise is not adequately captured, and the method has been shown in experimental trials by [158] to underestimate the covariance. The authors of [158] improved the work of [157], but the results still showed an underestimate of the covariance. Stein-ICP[159] can provide a better covariance estimation than previous methods, using a sampling-based approach but, therefore, also has a lower computational efficiency than closed form solutions. However, Stein-ICP can be parallelized to run on a GPU for optimal time efficiency. Stein-ICP proved to provide good covariance estimations and is therefore used as the method of choice for this paper.

Stein-ICP initializes a set of K randomly generated particles within some set 6D pose boundaries. For the use case with the UAV we can limit the initial distribution of the points on the roll and pitch axis, since we know these from the IMU of the onboard FCU. Using an accurate IMU, one could also exclude these two degrees of freedom; however, due to some uncertainty and noise in the IMU measurements on the UAV, we still include the roll and pitch degrees of freedom in our experiments. Each particle represents the transformation of sampled points from a source point cloud, which in our case is the sensor scan. For all points in each transformed batch, the corresponding closest point in a reference cloud (map) is determined based on the nearest neighboring points in the 3D space. Mean gradients are then estimated for all matching pairs belonging to each K particles. Next, the Stein variational gradients [173, 159] are obtained independently for translations and rotations, which, in turn, are then used to update each particle. This estimation is repeated for T iterations, producing an adjusted set of K particles that represent the posterior distribution. The distribution of the K particles can be used as a discrete estimation of the uncertainty and ambiguity of the environment for each degree of freedom in the registration. Using the Kernel Density Estimation (KDE) method on the distribution, the best registration can be selected. An inherent feature of Stein-ICP is that it can also apply a small refinement of the registration, which is utilized to obtain a more accurate local registration than the one provided by the global TEASER registration algorithm.

Registration quality After the uncertainty and refinement step have been performed, the quality of the registration is calculated. The quality estimation is based on the overlapping points between the map and the transformed sensor scan, where overlaps are considered if points are within the resolution of the voxel down sampling of the point clouds. The threshold for the experiments is set to reject registrations below 75% of overlapping points. This allows for some inconsistency between the map and the scan point cloud, which could be due to areas with severe buckling or structural revisions not applied to the CAD drawings of the vessel.

Visual Inertial Odometry

To aid the localization system with pose updates between absolute pose estimates, a relative pose system is introduced, namely Visual Inertial Odometry. The system in the experiments uses the front-facing Intel T265 camera, with a proprietary VIO algorithm, which provides a pose and velocity message. For our absolute localization system, we use the velocity message as the input to our Extended Kalman Filter.

Filtering

Outliers are removed from the absolute pose estimate using a motion filter that is based on a maximum velocity threshold of the pose estimate compared to the previous pose update. The filtered pose and covariance matrix is then used in an UKF[174, 175] together with the VIO pose and covariance matrix. The UKF is a non-linear filtering algorithm, and is well described in [174], and thus only a high-level explanation will be given here. It is based on the unscented transform (UT) technique for propagating a mean and its covariance through a nonlinear transformation, and the UKF is proposed as an improvement to the well-known linear EKF. Instead of the linearization required by the EKF, the UT approximate method is used in the UKF. A set of weighted sigma points are chosen based on the mean and covariance of a prior state. Each of these points are transformed to a new state using the non-linear function in the UKF. The predicted mean and covariance are then calculated on the basis of the newly transformed points. The added value of UKF compared to EKF is that it better represents non-Gaussian noise in the system[176].

Stein-ICP vs P2P-ICP

Figure E.7 illustrates a point cloud registration for a single scan using standard Point-2-Point (P2P) ICP[177, 150] and our 3D registration part of the pipeline. P2P-ICP settles at a local minima, with the y- and z-axis being correct, but with an incorrect registration of 1.8 meters on the x-axis, as can be seen in Figure E.8. With the current settings of Stein-ICP, p2p-ICP is more than 100 times faster than Stein-ICP, however the structural ambiguity of our environment results in p2p-icp setteling into the incorrect local minima. Besides ICP's large incorrect registration on the x-axis, ICP also fails to provide a covariance or uncertainty estimate of the registration, which is the main benefit of using Stein-ICP in this paper. Based on our experience, another benefit of Stein ICP vs. ICP is its ability to additionally provide refinement in slightly erroneous registrations provided by TEASER++.

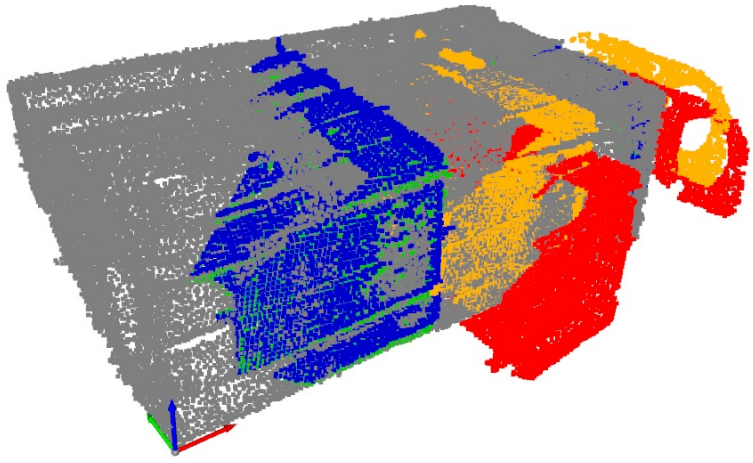


Figure E.7. Stein vs ICP point cloud registration. Red is the raw scan point cloud, green is the ground truth, yellows is p2p-ICP and blue is the Stein ICP pipeline.

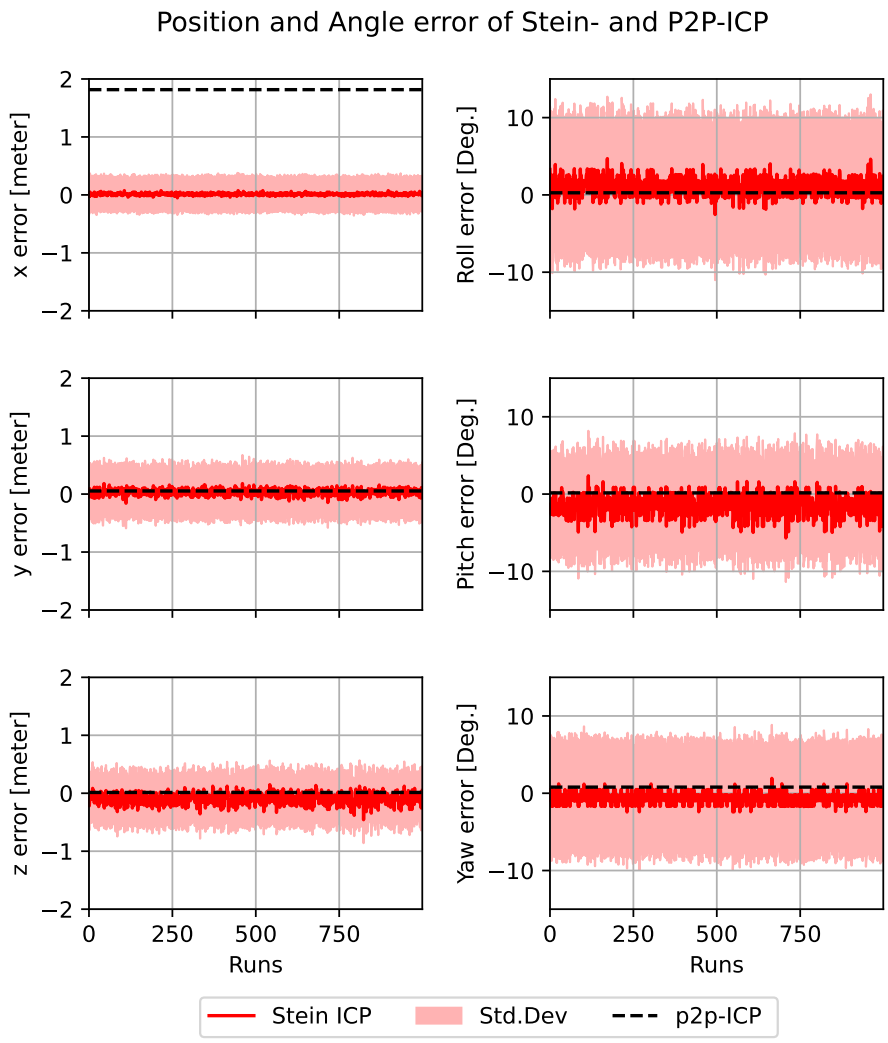


Figure E.8. The position and angle error of p2p-ICP registration (deterministic) compared to statistics (mean and STD) of stein ICP (1000 runs)

E.4 Experimental results

Due to limited access to real ships during the COVID-19 pandemic, experiments were carried out in a mock-up model of a water ballast tank. A series of flights with different inspection points and paths within the mock-up model were carried out. A single flight is described in this section.

The path of an inspection flight can be seen as a top view and a side view in Figure E.9, where the UAV enters the ballast tank through the man-way in the wall on the left. Hereafter, the inspection mission begins, with its first waypoint being (1,1,1). The UAV then visits the predefined interest points, marked as red stars, looking for defects at the inspection point itself and along its path, shown as the dotted black line. Figure E.9 also shows the estimation of the tags/defects illustrated as the blue points, where the equivalent ground truth is shown as a red circle. As can be seen in the figure, some position error in the estimate of the defect location is still present, where this error is shown as a box plot for each of the 6 defects in Figure E.10.

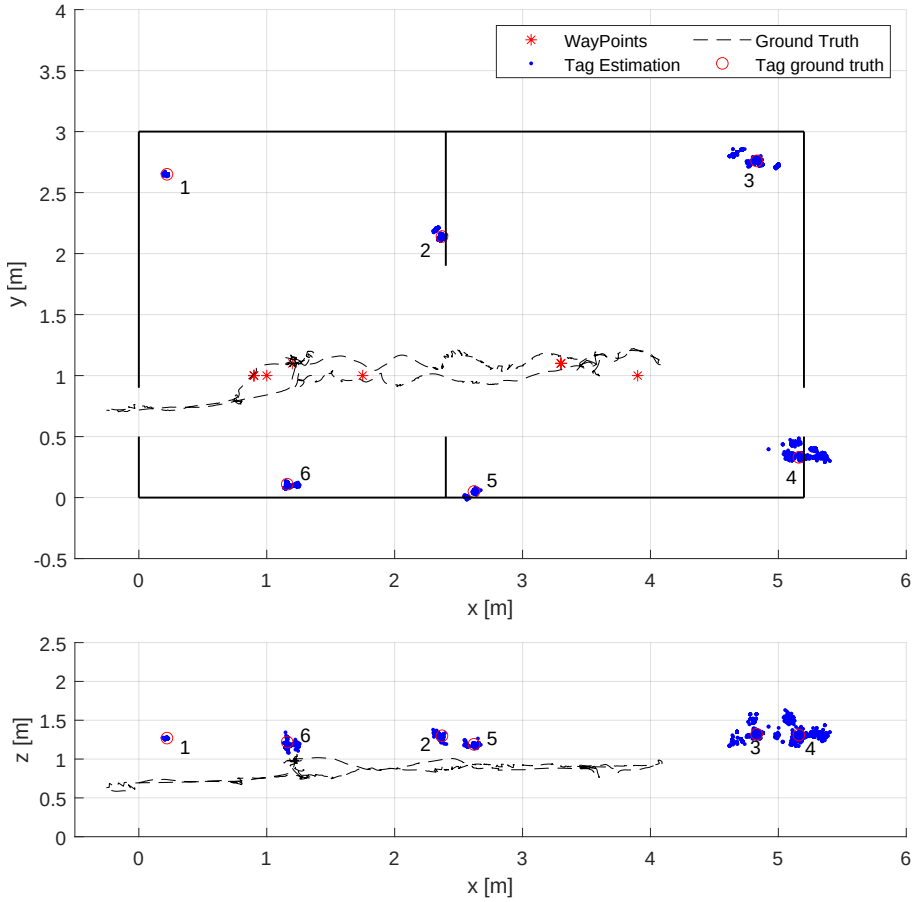


Figure E.9. Top down view of the trajectory of a single inspection run in the mock up model of the water ballast tank. The thick black lines indicate the walls of the tank. The dotted black line is the actual trajectory of the UAV. The blue points indicate the estimated position of defects and the red circles their actual position. The red stars mark the way-points the UAV has to visit on its inspection run.

From this figure it can furthermore be seen that the errors are generally low, but point 3 and 4 have a larger variation on the x-axis than the rest. This is caused by both an imprecise and varying timestamp provided by the L515, and the points being in the FOV even at large distances, which in turn exaggerate small camera calibration errors. An increasing error also arises during long time intervals between absolute updates from the localization system. The position of the specific points can be seen in figure E.9.

The error of the absolute localization pipeline can be seen in Figure E.11, where the blue line indicates the output of the entire pipeline.

The 3D registration estimate using FCGF, Teaser++ and Stein-ICP is shown as red stars, and depicts the absolute error of the pure 3D registration element in the pipeline. All 3D registrations are estimated every 2 seconds (0.5 Hz), whereas the UKF, which fuses the VIO with the 3D registration pose estimate, runs at 30 Hz. The intermittent output from the Teaser++ registration algorithm, in Figure E.6, using either SmoothNet or FCGF as its feature descriptors, are shown as the purple and black crosses, respectively. The SmoothNet feature descriptor was unable to run reliably in real time on the Jetson NX, and was therefore subsequently added using a ROS-bag file on a desktop PC. The FCGF feature descriptor was used in the pipeline during the flight as shown in the figure. The

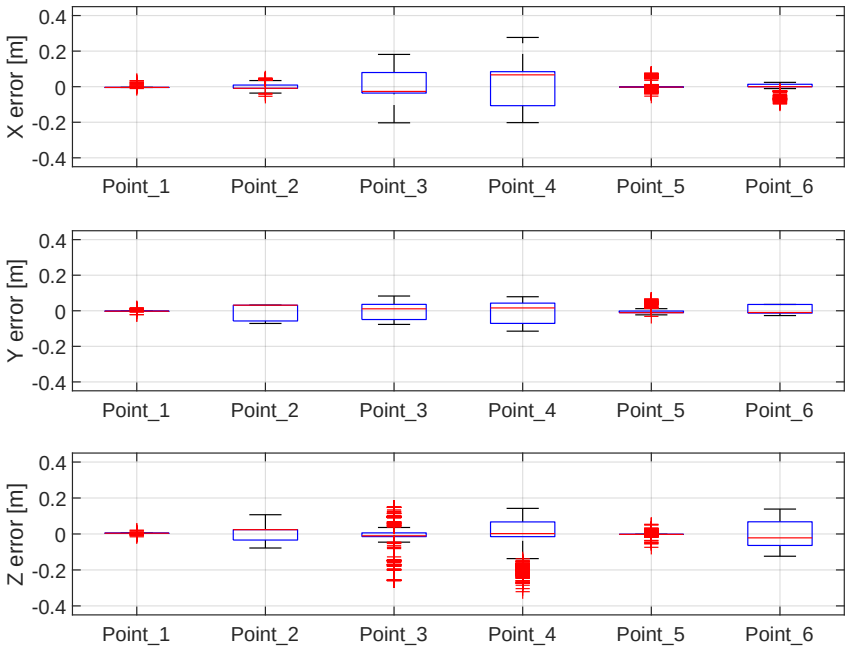


Figure E.10. Box blot of the position error for each of the 6 defects located in the tank

results from [3, 4] however indicate that SmoothNet would be slower than FCGF by more than a factor of 10 if running on the Jetson. The computational time for each absolute pose update is on average 1.85 seconds. The mean accuracy and standard deviation of each feature descriptor, together with the output of the filtered output pose, can be seen in table E.1. From the table it can also be seen that Stein-ICP corrects some of the x-position error from the raw FCGF-Teaser++ output, as was also shown in Section E.3.2.

The vertical dashed lines indicate an absolute pose update of the UKF from the 3D registration part of the pipeline. The varying intervals between the lines is due to the rejection of low-quality 3D registrations based on the overlapping points between the map and scan.

To estimate the ground truth of the UAV, a series of closely spaced April Tags are mounted at known locations on the floor of the ballast tank. The ground-truth tags are only mounted on the floor in order to avoid providing artificial visual feature points for the forward looking VIO camera. Using the image stream from the downward facing T265 camera, the localization part of the TagSLAM package[171], can be used to provide a ground-truth estimate. The downward-facing camera data and the derived ground-truth estimate are not used in the localization pipeline but only serve as a method for comparing the performance of our absolute localization estimate. Due to the limited space inside the ballast tanks, a motion camera system such as VICON was not deemed a viable solution as a ground-truth estimate. To represent the defects within the tank, six AR tags are mounted on the walls. Tags are recognized using the image stream from the L515 color camera, which serves as the inspection camera of the UAV. When a tag/defect is detected, the inspection system will save a point cloud and image of the defect and estimate the defects absolute position in the map frame.

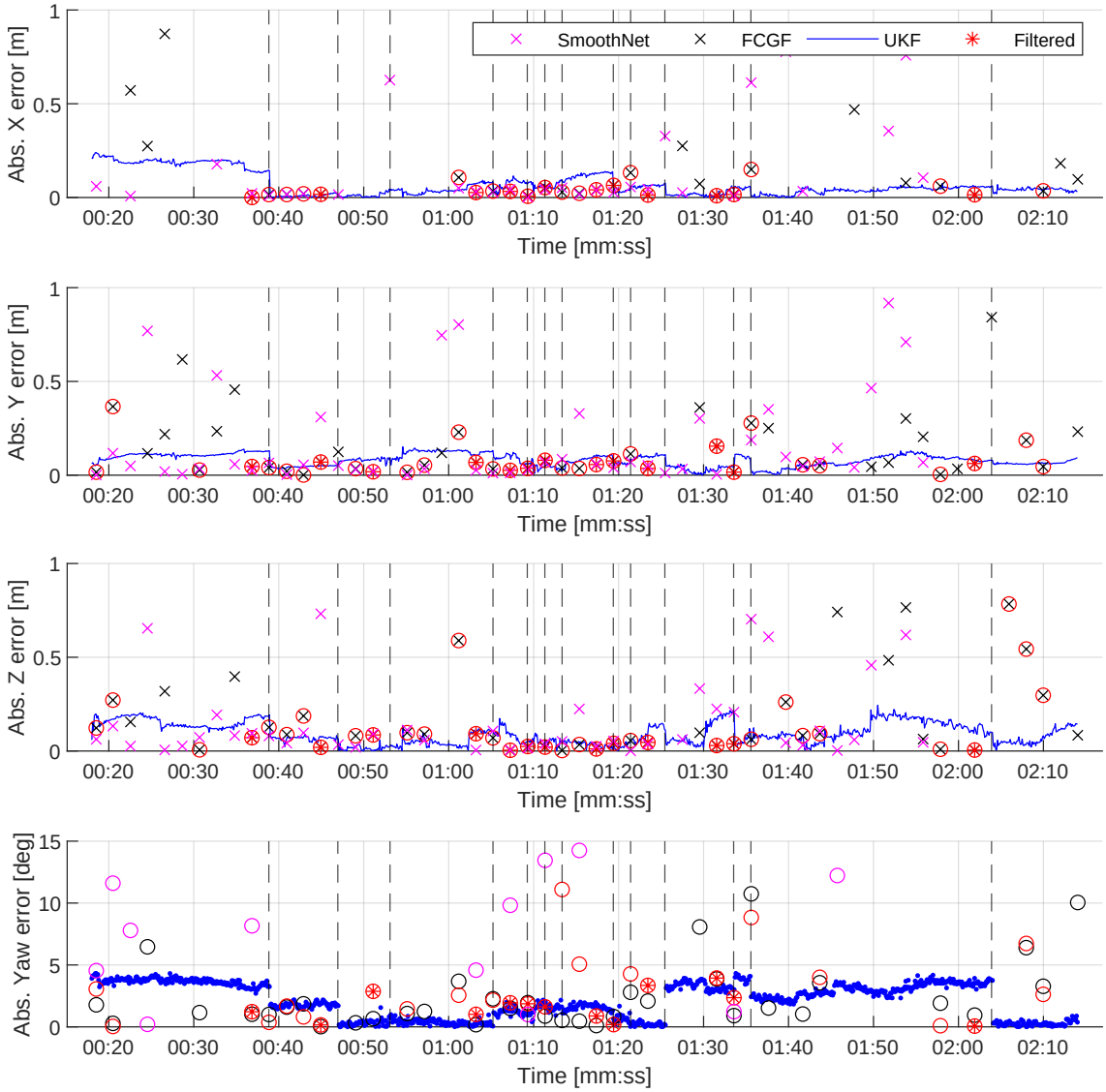


Figure E.11. Error of localization pipeline. Vertical dotted lines indicate an absolute pose update with a correction to the position of the robot.

Table E.1. Mean and standard deviation of absolute pose estimations using different feature descriptors

Mean [m]	3DSmoothNet	FCGF	FCGF+Stein	Filtered
x	-0.67	-1.09	-0.60	-0.24
y	-0.08	0.13	0.13	0.04
z	-0.08	-0.01	-0.06	-0.02
Std. Dev [m]				
x	1.43	1.86	1.72	0.80
y	0.47	0.69	0.51	0.04
z	0.70	1.03	0.21	0.04

E.5 Conclusions

In this paper an autonomous system was designed and tested to inspect known confined spaces within a mock-up model of a water ballast tank. Our method used state-of-the-art deep learning-based feature descriptors on 3D point cloud scans from a time-of-flight camera. The features were registered to a know map of the environment using machine learning based correspondence search and the fast Teaser++ registration algorithm, to provide an absolute pose estimate within the environment. The novel GPU-based uncertainty estimation method, Stein ICP, was used to provide a 6-dof uncertainty estimate for the highly ambiguous and structured environment. The system was tested in real time on a small UAV-platform without the need for external human input during the inspection execution. Using 3D registration and Visual Inertial Odometry, the UAV was able to accurately navigate and inspect the featureless and ambiguous environment of the water ballast tanks and, moreover, locate and estimate the position of installed defects. The position of the defects were accurately reported within the map frame to be used for further evaluation by the human surveyor.

Future improvements to the system would be to improve the feature descriptors and the registration algorithm to better handle the high level of ambiguity in the environment. Automated detection and evaluation of defects is also an addition that would further add additional value to the overall inspection system.

Article F

Absolute Localization in Feature-poor Industrial Confined Spaces

Rune Y. Brogaard[1], Robert A. Hewitt[2], Sarah Etter[3], Arash Kalantari[2], Evangelos Boukas[4]

[1]FORCE Technology, Park Allé 345, Brøndby, Denmark
[2]NASA Jet Propulsion Laboratory, Pasadena, CA 91109, USA
[3]University of Southern California, Los Angeles, CA 90007, USA
[4]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

Abstract:

Autonomous inspection of dark, confined, and feature-poor spaces requires robotic platforms to utilize accurate and reliable localization systems for safe and reliable operation. This paper presents an absolute localization system for highly feature-poor spaces, using visual inertial odometry and GPU-based point cloud registrations for limited field-of-view sensors. The extracted structural elements from sensor scans, along side IMU measurements, are used to limit the search area for the GPU-based point cloud registrations. We employ Stein-ICP which is an uncertainty aware variant of ICP. The 3D registrations are then fused with a visual-inertial odometry estimate in an Extended Kalman Filter to provide a fast and accurate absolute pose estimate. The proposed localization system is tested in both a simulated environment and in a mock-up model of a chemical distillation column — both highly feature-poor areas.

F.1 Introduction

Localizing within dark, confined industrial spaces can be a challenging task in robot applications. Structured areas, such as a distillation column, often contain featureless surfaces, limiting visual tracking or recognition. Repetitive patterns —of both visual and structural nature— can make loop closure, and by extension absolute localization, prone to errors. Furthermore, occlusions and reflections can diminish the accuracy of visual-inertial odometry (VIO) systems, while low-light conditions limit the number of reliable visual landmarks. Additionally, environmental attributes, such as dust or moisture, can introduce noise in the sensor data and therefore also in the pose estimation. Data association of 3D

feature descriptors, such as FPFH[135], FCGF[138] and 3DSmoothNet[136], are prone to incorrect matching due to repetitive patterns in an environment. Generic ICP requires a good initial prior, without which it can converge to an incorrect local minima of the cost function. This can be especially problematic for ambiguous cases or where the initial prior has a high uncertainty. Due to the similarity of our environment —a symmetrical metal column— we are in need of a localization system that can handle somewhat ambiguous and feature-poor cases. An example of an ambiguous case is shown in Figure F.2, where red points indicate the registered scan from the sensor. In this example, it is clear that the scan can be reasonably registered to multiple locations within the cylindrical map. The main contributions of the paper are:

1. Incorporating the uncertainty-aware 3D registration algorithm Stein-ICP on an online robotic system for EKF-based absolute localization.

2. The improvement of Stein-ICP based estimation by using the Kernel Density Estimation distribution evaluation.

3. Taking advantage of prior information about the environment and robot to both speed up Stein-ICP by sampling based on state uncertainty and to reduce scene ambiguity by only registering local views to unique elements of the environment.

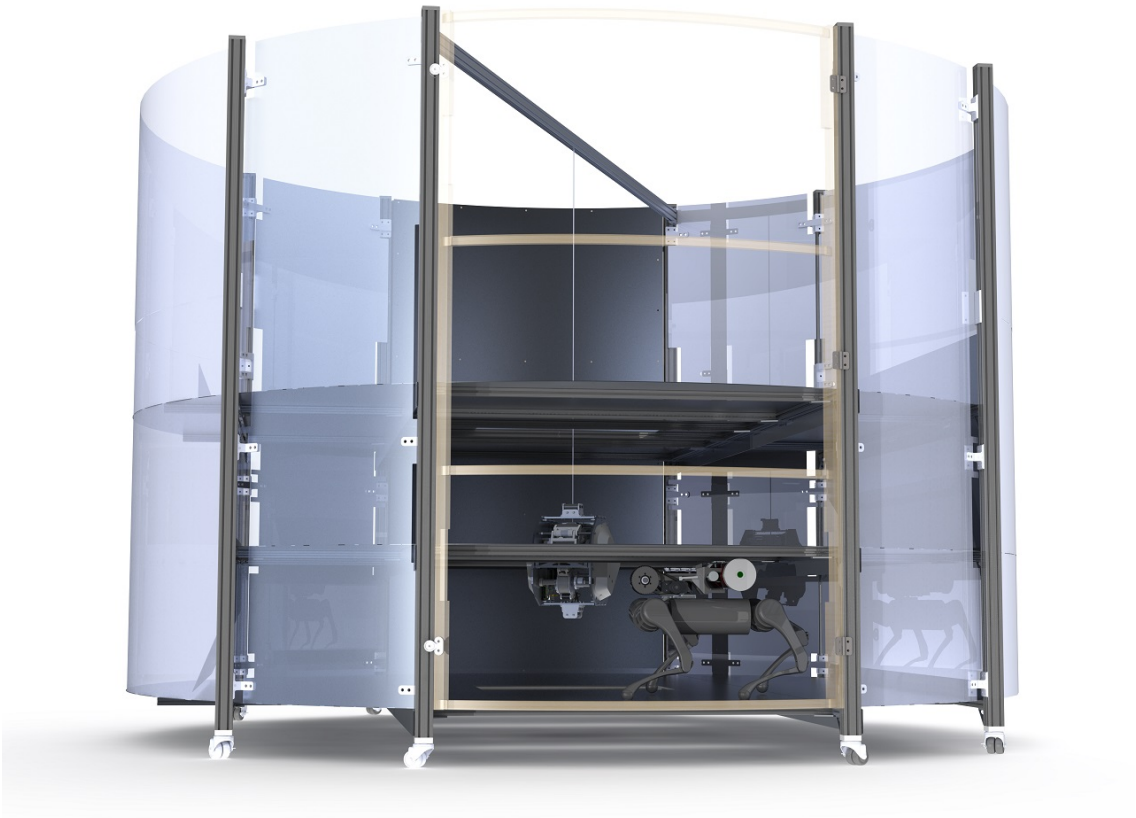


Figure F.1. Mock-up model of 3 sections of a distillation column with a diameter of 3 meters. The model is based on a real world column and is used for the testing of our absolute localization pipeline.

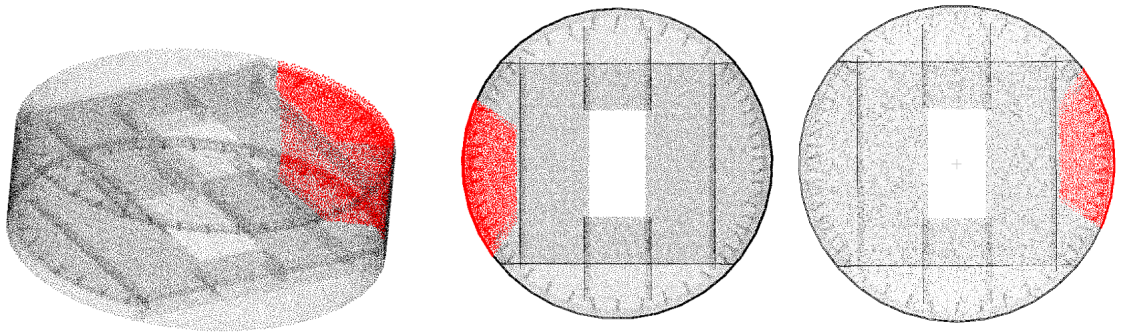


Figure F.2. Example of ambiguity in a single section of a typical distillation column. The grey points represent the map and the red points a point cloud from the sensor. The red point cloud can reasonably be registered to multiple locations in the map, from which the ambiguity problem therefore arises.

F.2 Related work

Visual localization systems are a good option for legged robots, but are known to produce at best 1% error [122], [142] which is also unbounded without some prior knowledge or re-observation. Although field robots can rely on distinct landmarks for absolute localization [145], a featureless structured environment, such as our distillation column, can cause increased errors and can lead to failure. Furthermore, low light conditions and reflections from onboard light sources, can limit the number of reliable visual landmarks. Furthermore, dust and high humidity can introduce noise to the sensors, which therefore affects visual pose estimation. Therefore, additional steps are needed to provide a reliable localization estimate.

The authors of [143] use a VIO pose as a prior for an ICP registration, and add further validation and filtering of the registration result. However, this only mitigates the issue of incorrect ICP registrations by filtering the outliers, instead of reducing the number of incorrect registrations.

The research in [178] also uses an ICP approach, and assumes known rotations for the roll, pitch and yaw angles. The authors used the eigenvectors associated with the Hessian of the Point-To-Plane ICP cost to describe the observability for each direction for a pose estimate. This method can provide an estimate of how well the registration can be performed on each of the x, y, and z axes, which reduces the uncertainty in the pose estimation.

The researchers in [14] investigate the possibility for absolute localization using a pre-installed Ultra Wide Band(UWB) satellite system coupled with a VIO pose estimate. Although this setup can produce good estimates, it is only suitable for frequently inspected areas, such as production lines or warehouses. It is therefore clear that for most industrial inspection cases, this solution is not suitable for absolute localization, which is also the case for robots operating within our distillation column.

The use of geometric structural features for absolute localization in confined spaces has previously been investigated in [2], however, the feature descriptors were designed to represent structural attributes that are use case specific.

The rise of fast and accurate 3D descriptors, such as [136], [148], [138] and robust registration approaches [140], [141], [149], provides a good toolbox when trying to tackle absolute localization. The authors in [4, 3] use novel machine learning-based feature descriptors and state of the art registration algorithms, however the approach still suffers from incorrect data association in ambiguous environments. A motion filter is used to mitigate this issue, but the underlying 3D registration outliers still persist.

F.3 Environment

In our use case, a quadrupedal robot has to localize itself within a single section of a chemical distillation column. These columns are large cylinders, often more than 20 meters tall, with varying diameters of 1 to 4 meters, and with a set of trays/levels inside the cylinder. Due to this cylindrical shape and the man-made structure, very few unique geometrical features are available to be described by feature descriptors such as FPFH and FCGF.

To enable rapid development and testing, in addition to a simulation environment, a mock-up model of a distillation column was built. The model is based on a sub-section of a real distillation column and consists of two levels, as shown in Figure F.1. The aluminum sheet metal chosen for the walls of the mock-up model was too reflective compared to real-world inspection cases, and the walls were therefore covered by a less reflective material in the experiments.

The robot was placed inside the mock-up model for the real-world experiments. The ground truth was extracted using an upward facing camera —specifically an Intel D455 which is not part of our system— and ARuCo markers on the ceiling of the column. To estimate the positions of the tags, which serve as ground truth markers, TagSLAM[171] was used. For the online ground truth estimation we used AprilTag 2[170] on the images from a top-mounted camera (D455).

F.4 System description

F.4.1 Localization

The localization system, shown in Figure F.3 is divided into three modules: 3D registration, VIO, and Filtering. However, before the pipeline is started, the system must first load the map of a single section in the column. This is the map that the 3D registration module (highlighted in orange in Figure F.3) of the pipeline uses to estimate the pose of the robot. The 3D registration module handles the 3D point cloud registration, where a point cloud recorded by the sensor is matched to the correct location in the map. The 3D registration module consists of four sub-modules. The first sub-module in the pipeline is the structure detection block, which classifies and labels the specific elements in the column based on their geometric properties. Examples of these elements are the walls, floor, ceiling, and the downcomer in the column. The walls can be fitted to a cylinder using a model-based RANSAC apporach, and the center point and radius of the cylinder can be extracted and used later in the pipeline. The structural elements are also used as indications of whether our system has enough structure in the scan to accurately estimate a position. For

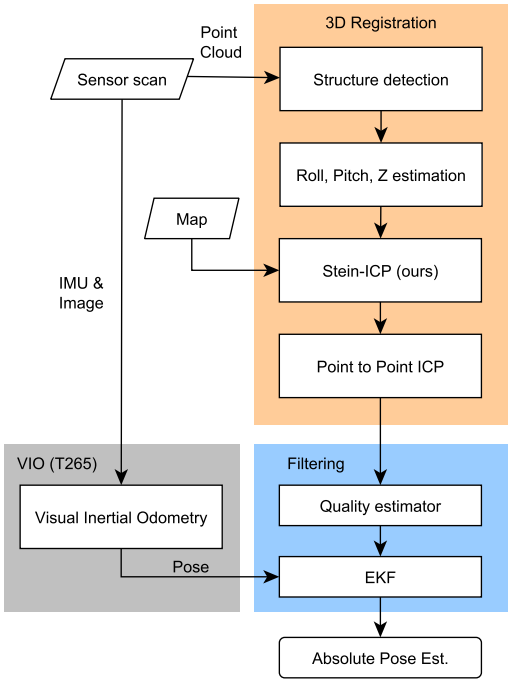


Figure F.3. The absolute localization pipeline consists of 3D registration, VIO and a filtering step, which fuses the two pose estimates in an Extended Kalman Filter.

the specific case of the distillation column, the downcomer detection is used as a simple indicator that a good registration should be possible. Based on the segmented wall points, the structure detection module estimates the diameter and x, y center of the column. If the estimated diameter is equal to the specified column diameter the initial transformation of scan point-cloud is initialized so its x, y, z-origin is aligned with the map's x, y, z origin. The z-value corresponds to the elevation of the camera above the floor, given by the joint poses of the robot's leg when standing on the column floor section. The roll and pitch angle of the camera, from its onboard IMU, is furthermore used to initialize the roll and pitch rotations for registration estimation. The actual registration between the scan and map point cloud are done in the next step, employing Stein-ICP. This sub-module is a 3-dof optimized version of the standard 6-dof Stein-ICP algorithm[159]. The last step in the 3D registration module is a point-to-point refinement step, using a generic ICP registration (Open3Ds[150]).

The second module of our system (highlighted in gray in Figure F.3), constitutes the relative localization of our robot and can be provided either by means of dead reckoning (eg: the quadrupedal odometry), or by more robust techniques. Examples of such techniques are the multilevel-patch and EKF-based visual inertial odometry (ROVIO [25]) or the multi-frame tracking VIO methods such as the ones used by the Intel Realsense T265 (which is the camera we employ in this work). The last module (highlighted in blue in Figure F.3) constitutes a loosely coupled EKF, which fuses the 3D registered pose estimation with the VIO estimate to increase the update rate of the resulting absolute pose estimation. This module only accepts high quality 3D registrations, due to the quality estimation sub-module, which qualifies 3D registrations based on the their average RMS error between the registered scan and the map.

F.4.2 Structure detection

The first important step in our localization pipeline is to detect and label the structures in the camera field of view. An example where we label the column elements from the scan can be seen in Figure F.4. The red points indicate the floor of the current level, while the green points indicate the cylindrical wall. The yellow points represent the downcomer of the most common and convetional crossflow tray[179]. The presence of the donwcomer indicates that the robot is at a location to facilitate a high quality registration between the scan and map, since only one downcomer is present on each tray. There are many approaches that can be followed to identify such elements, for example, deep 3D shape detection [180] or even structural landmarks [77] can be employed. In our case, for the sake of simplicity, we use the open source Point Cloud Library (PCL) where we label points in the cloud, based on core 3D shapes such as cylinders and planes. The different shapes are extracted using a model-based RANSAC approach. The output of the cylindrical model fitting is the x,y center point of the cylinder, which we use as the initialization step of our Stein ICP algorithm.

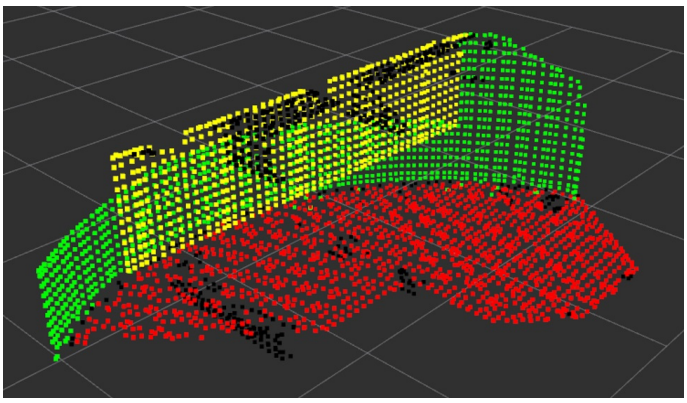


Figure F.4. Classification of the structural elements in a sensor scan from the simulated column. The floor is depicted with red points, the walls of the column as green points, and the downcomer as yellow points. Black points indicate unclassified points.

F.4.3 Stein-ICP and Filtering

Since our absolute localization system relies on a Kalman-based state estimation submodule, specifically the EKF, it is of utmost importance that our state and measurement covariances are well defined. Assuming the world frame F_w and a robot frame F_r, the state of the robot is notated as $x_w^{r_k}$ (which is also our absolute pose), at time k with P_k uncertainty:

$$x_w^{r_k} := \left[p_w^{r_k}, q_w^{r_k}\right], \tag{F.1}$$

where $p_w^{r_k}$ is the robot position and $q_w^{r_k}$ is the robot unit quaternion which represents the rotation from frame F_w to frame F_{r_k}. Additionally, $x_{r_{k-1}}^{r_k}$ denotes the local state (relative localization) between 2 consecutive image/3D frames of our robot. During the prediction and update iterations of the EKF, both the predicted $\hat{x}_w^{r_k}$, $\hat{P}_k$ and corrected $x_w^{r_k}$ and P_k states and covariances are employed. Our system employs both a relative pose estimate (coming from the T265) and an absolute pose estimate as a byproduct of our 3D registration (using the Kinect sensor). While the projection among different sensors is trivial, the definition of the absolute localization covariance as input to the EKF is not.

A typical approach for 3D-based absolute localization is the Iterative Closest Points (ICP) algorithm. ICP [181] has several "flavors" based on the data used to perform the point clouds alignment — i.e. point-to-point, point-to-plane, plane-to-plane. The solution to ICP is an optimization, often solved using Singular Value Decomposition (SVD) and has the following form:

$$\underset{R_o,u_o}{\operatorname{argmin}} E(R,u,s_i,r_i) = \frac{1}{N}\sum_{1}^{N} ||(Rs_i+u)-r_i||, \tag{F.2}$$

where E is the alignment error of N corresponding points in clouds r_i (map) and s_i (robot view), given a rotation matrix R and a translation vector u.

While most ICP approaches do not provide an uncertainty for the alignment, and are therefore not suitable for our method, some proposed variations can estimate the covariance of the transformation [157, 158].

As most approaches either over- or under-estimate the covariance, for the actual registration of the map and the scan point cloud we use a modified version of the GPU-based Stein-ICP[159] algorithm, as it has been proven to provide accurate uncertainty estimates. Stein-ICP initializes a set of K randomly generated particles, within some set 6D pose boundaries. Each particle represents the transformation of a batch of sampled points from a source point cloud (scan). For all points in each transformed batch, the corresponding closest point in a reference cloud (map) is determined based on the points nearest neighbour in 3D space. The corresponding points are stored in pairs, and their mean gradients are estimated for all matching pairs. Next, the so called Stein variational gradients are obtained independently for translation and rotation, which are then used to update each particle. This procedure is repeated for T iterations producing K particles representing the posterior distribution.

Our version of Stein-ICP has two distinct modifications. Firstly, we have set up Stein-ICP to not optimize the Z, roll, and pitch degrees of freedom and is, therefore, able to run faster than an equivalent 6D pose optimization. Our motivation to create this minimal version of Stein-ICP is that the estimate of the scan point cloud Z-position can be obtained from the Unitree robot, given our prior knowledge of the column structure and the robot's joint encoder measurements. The uncertainty of these states are obtained form the Unitree robot own estimate. Likewise, the roll and pitch component can be obtained from the onboard IMU, maintaining the manufactures uncertainty estimate, eliminating the need to estimate these rotations as well. The system is therefore reduced to optimizing only on the x, y, and yaw components. The second difference compared to the original Stein-ICP —as it was also hinted in the original paper— is that the initialization of the K points is not random, but stems from our structure detection (see section F.4.2). Specifically, the K points are defined by the output of multiple ICP runs. Each ICP run is initialized at the center of the structure detection solution but with different yaw (while multiple ranges were tested and work adequately, for this work we will use uniformly distributed yaw values in the range $[-\pi,\pi]$). These ICP iterations come at a very low computational cost as they are GPU accelerated. The average compute time for our setup, with 200 particles, is 3.3 seconds using the original Stein-ICP and 2.3 seconds for our reduced version. The structure detection part of this time consumption is on average 0.5 seconds. Even though the 2.3

seconds are lower than the framerate of our sensor and frequency of estimation, it is used only for correction, while VIO is used in the interim. After the optimization, the output of our Stein-ICP is a distribution of translations on the x and y axis, and a distribution on the yaw(z) rotation. While the Stein-ICP output distribution provides a valuable uncertainty estimation, the assumption of a normal distribution can lead to erroneous results, as the mean solution can be relatively far from the true pose. Therefore, the final translation and rotation is selected as the value with highest density from a Kernel Density Estimation for each degree of freedom. An example of how the yaw rotation angle is selected is shown in Figure F.5. From the plot it can be seen that using our KDE approach —instead of the standard normal distribution— has an impact of a 44 degrees difference. Our KDE top-point is located at 48 degrees, where the ground truth is at 48.7 degrees. Finally, the x,y, and yaw of the VIO system are fused with the 3D registration pose in an Extended Kalman Filter[175].

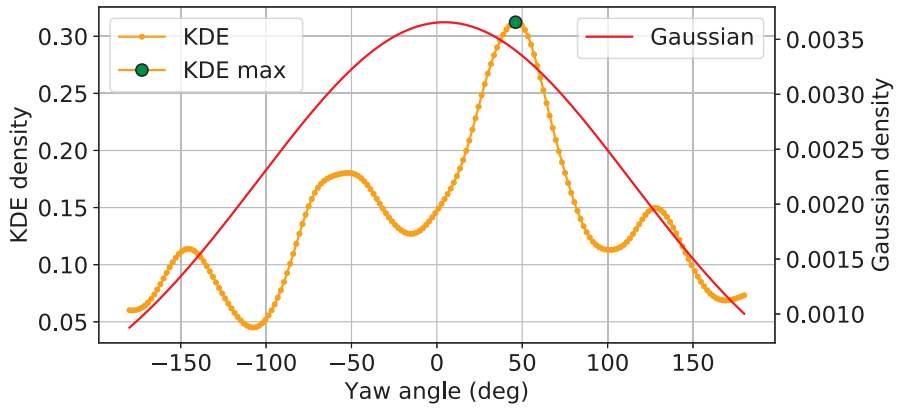


Figure F.5. Example of how the yaw rotation angle in the 3D registration is determined based on the Kernel Density Estimation (KDE) method from our modified stein-ICP approach. The standard Gaussian distribution is plotted based on the mean and variance of the stein particles. The difference between the KDE and Gaussian top points are 44 degrees.

F.5 Experiments

To test the absolute localization pipeline, a simulated environment and a mock-up model of a distillation column were created. The design of the mock-up model, shown in Figure F.1 was based on a subsection of a real-world distillation column. For realistic comparison, the distillation column used in the simulated environment is based on the same CAD model as the mock-up model. A separate PC is used to run the Gazebo simulation environment. The map used for both the simulation and real-world tests is a point-cloud version of the CAD model.

F.5.1 Robot

To inspect the distillation column, the Unitree A1 legged robot, shown in Figure F.6, was chosen —based on its small size and its agility compared to other wheeled solutions. It is equipped with a Kinect Azure ToF camera and an Intel T265 camera. In simulation, the robot is furthermore equipped with a custom robotic arm used for transitions between the different trays in the distillation column. By using an accurate depth and color camera,

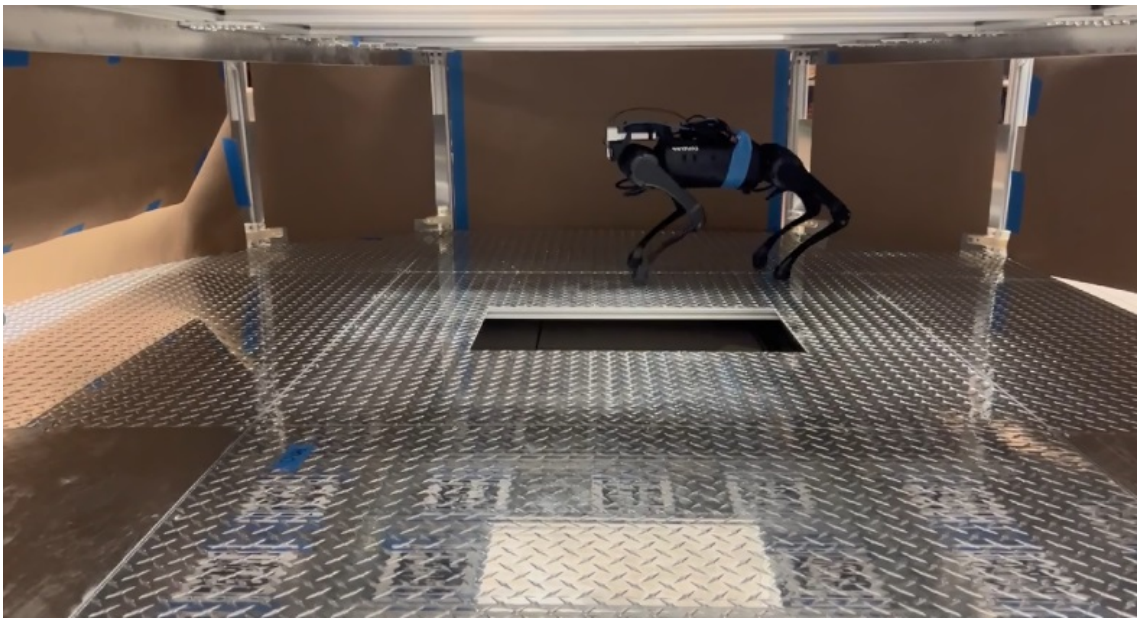


Figure F.6. The Unitree robot standing inside the mock-up model used in the experiments. The Kinect Azure and T265 camera can be seen mounted on the front of the robot.

the robot is able to collect both color images and colored point clouds of the interest points for inspection. This is especially useful for inspections where a 2D image might not be sufficient to detect buckling in the structure or similar visual hidden structural defects. For computation and low-level control, the robot utilizes an Intel NUC i7 8th gen., which is supported by an Nvidia Jetson Xavier NX handling the localization pipeline and point cloud processing in realtime. A D455 camera is furthermore added pointing upwards to capture ArUco makers added to the ceiling of the tray above the robot. This camera is not used in the localization pipeline but only as a method to capture our ground truth of the robot, when operating inside the mock-up model. The localization pipeline is implemented using the ROS framework, and the filtering is all performed in the base link frame, which is located in the center of the body.

F.5.2 Simulation

The Unitree robot and distillation column were simulated using ROS and Gazebo. The parameters of a generic point-cloud camera in Gazebo were adjusted to fit the specifications of the Microsoft Kinect Azure camera. To emulate our VIO pose estimate, a drifting odometry was simulated based on the pose of the robot in the gazebo simulation. The robot was manually tele-operated to simulate an inspection run inside the column. A top view of the path is depicted in Figure F.7. From the figure it can be seen that the robot moves around the centered man-way on the column tray. The absolute x, y, and yaw-error of an simulated inspection routine are shown in Figure F.9a. At 31 seconds, pose updates (red stars) from the 3D registration module of our pipeline stop being provided. This is because the 3D registration module can no longer extract structural elements from the scan, since the camera is too close to the wall of the cylinder to generate suitable point clouds (downcomer not in view). An example of such a point cloud is shown in Figure F.2. At 53 seconds the robot has moved to a position where 3D registration is again possible and our 3D Stein-ICP pose updates are re-established. Between these two time stamps the

filter is only relying on the drifting odometry, which is indicated by the increasing error shown in plot F.9a. After time stamp 00:53, it can be seen that the error is reduced when a new absolute pose update arrives to the filter. The errors for the simulation are shown in table F.1.

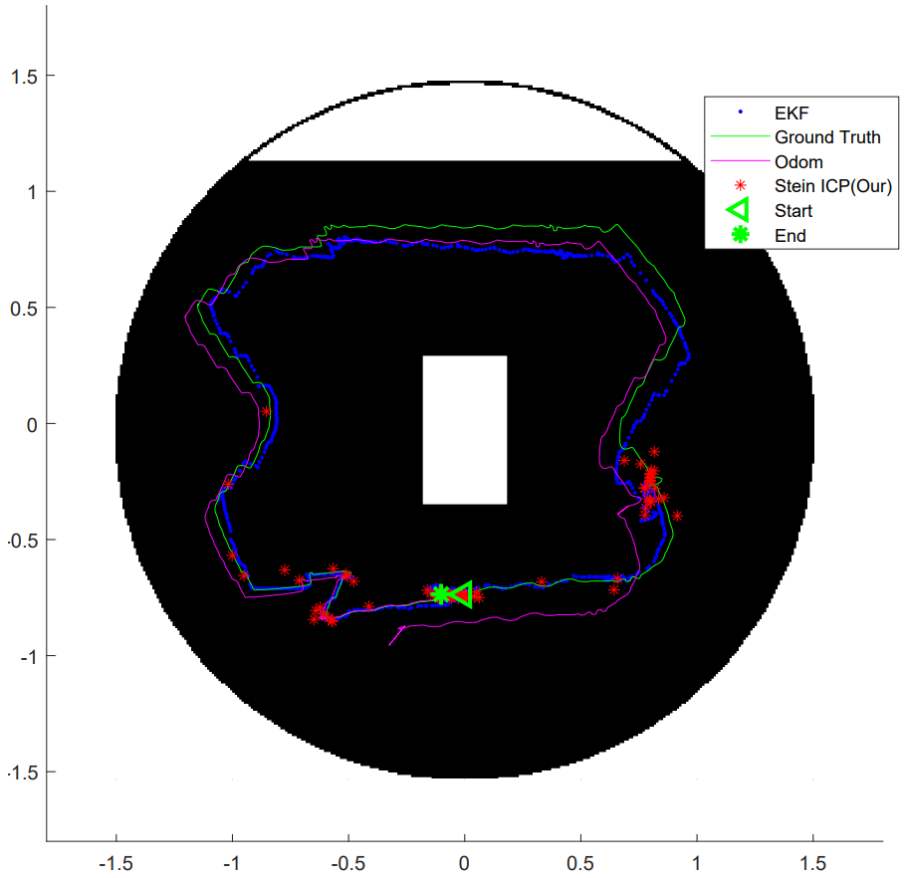
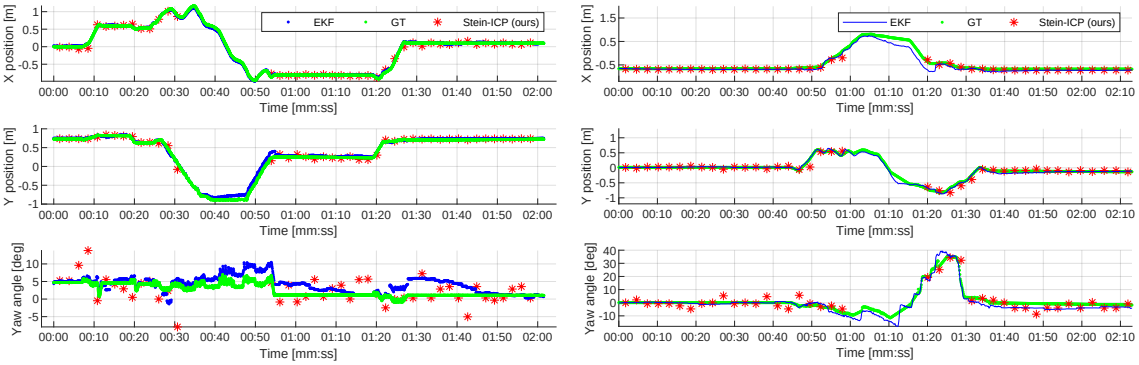


Figure F.7. Top-view x,y plot of the path of an inspection run in simulation. The green line indicate the ground truth, magenta the drifting odometry, the red stars are our Stein-ICP registrations and the blue dots is our absolute localization estimate from the pipeline. The white area illustrate empty space and the black area is occupied space.

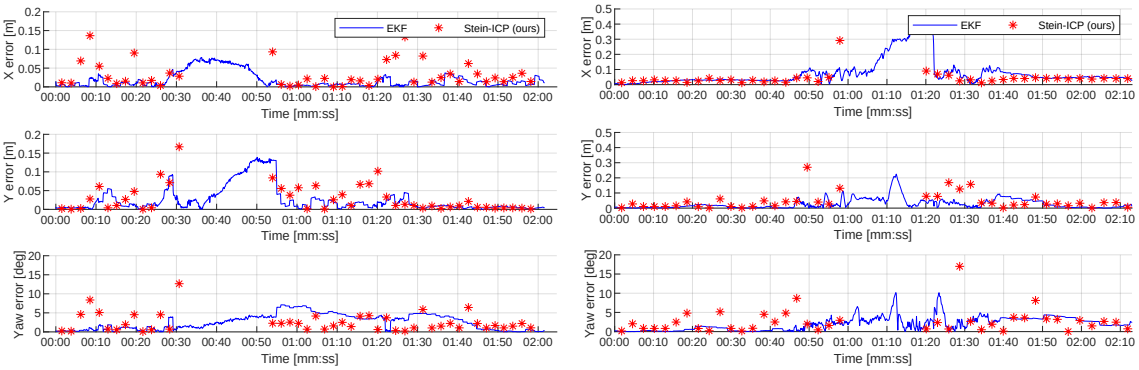
Table F.1. Error metrics for simulation experiments

Raw 3D registration:	***x [m]***	***y [m]***	***Yaw [deg]***
Mean	0.001	-0.006	-0.013
Std. dev.	0.042	0.0478	3.504
95 percentile	0.074	0.079	5.029
Max	0.136	0.188	12.350
Whole pipeline:			
Mean	-0.002	-0.029	-0.041
Std. dev.	0.024	0.040	2.211
95 percentile	0.053	0.018	2.437
Max	0.063	0.158	5.967



(a) Simulation ***(b)*** mock-up model

Figure F.8. Plot of the x,y position and the yaw rotation of an inspection run in simulation (a) and a similar run in the real world mock-up model (b). The output from the raw point cloud registrations are shown as the red stars: Stein-ICP (ours), and the ground truth is shown as the green line. The output from our localization pipeline is depicted as the blue line (EKF).



(a) Simulation ***(b)*** Mock-up model

Figure F.9. The absolute error of an inspection run in the simulated and real world mock-up model. The raw 3D point cloud registrations are shown as the red stars, whereas the output of the full absolute localization pipeline is shown as the blue line (EKF). The 3D registration drop-outs are due to lack of geometric features, which results in high dependence on the drifting VIO estimate.

Table F.2. Error metrics for real-world experiments

Raw 3D registration:	***x [m]***	***y [m]***	***Yaw [deg]***
Mean	0.032	0.004	0.207
Std. dev.	0.050	0.071	4.075
95 percentile	0.058	0.162	4.815
Max	0.291	0.269	9.891
Whole pipeline:			
Mean	0.073	0.013	1.359
Std. dev.	0.086	0.041	2.305
95 percentile	0.305	0.076	4.268
Max	0.379	0.224	10.158

F.5.3 Mock-up model

Our algorithm was also tested in the real world using a mock-up model of the distillation column. The approach follows the same steps as in simulation, where the robot was placed inside a section of the column and was manually controlled around the model in an inspection pattern. Due to the limited space, the ground truth was estimated using ArUco markers attached on the ceiling. An Intel D455 camera pointing upwards was used to give estimates of the ground-truth pose when the tags were in view of the camera. A plot of the x, y position and the yaw angle is shown in Figure F.8b. The absolute x, y, and yaw-error is shown in Figure F.9b and in table F.2, where it can be seen that the error is comparable to the simulated results. From the figure, it can be seen that at 58 seconds our algorithm can no longer perform 3D registrations, and the localization relies on the VIO estimate up until 1:20 where a new absolute pose is received. In this time span, it is clear that a noticeable error has occurred on the x axis and, to some extent, also on the y axis, due to the challenging environment for the VIO system. However, it is worth noticing that, with the exception of a few outliers, most of the 3D registration data points have small errors. This shows that even with a poor VIO estimate the algorithm is able to recover when structural data are available.

F.6 Conclusion

Within this work, we have designed and tested an absolute localization pipeline, using GPU parallelized 3D registrations, for environments with a low number of features and a high level of similarity. To incorporate 3D registrations into our state estimation, we employed a variant of the ICP algorithm, which is capable of providing accurate uncertainty estimates. Structural elements from the environment were detected and classified, to reduce the degrees of freedom of Stein-ICP. Initial results indicate good performance, with very few outliers for both the 3D registrations and the filtered output. The localization system was tested in both a simulated and a real-world environment and showed good performance in handling the highly symmetrical environment.

Acknowledgment

This research was supported by Dow Chemical, project 227027AT. This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the JPL Visiting Student Research Program (JVSRP) and the National Aeronautics and Space Administration (80NM0018D0004). This research was also supported by the Inspectrone project at DTU, funded by IFD Denmark. We would also thank Malcolm Tisdale from the Jet Propulsion Laboratory for designing and assembling the mock-up model.

Chapter 6

Applications of absolute localization in autonomous inspection

So far we have assumed that we operate in a known environment. This is a well-founded assumption due to the availability of simple CAD drawings or 3D models of most industrial confined spaces. However, in the cases where these models are not available, an exploration-based inspection approach could be used. An example of such an approach has been investigated in **article G**, where the Next-Best-View (NBV)[182] planner has been augmented with information derived from the defect classification. The goal of the original NBV planner is to explore the space, by generating random trajectories that maximize the path towards unseen space. The planner distinguishes between occupied space, free space, and unexplored space. The space is represented by an octree-based map[183]. A fixed number of nodes are sampled in the free space around the robot, using a Rapidly exploring Random Tree (RRT)[184], where each node represents individual states of the robot. These nodes are connected by a tree structure and are also given a score based on how much unexplored space is visible by the depth camera of the robot. The robot will move to the node with the highest score, and a new tree is then built from this node, using the maximum exploration approach.

In the article, the score of each node is furthermore scaled, based on the number of visible interest points. The interest points are continuously discovered by using a Faster R-CNN, trained on corrosion images generated in Blender[185]. By scaling the score of each node based on the visible interest points, gives the system precedence to trajectories that maximize the number of interest points in view, e.g. corrosion. This results in more images of the defects, and at different angles, and thereby gives a better foundation for the evaluation system – be it automatic or human-based. The main contribution of the paper is therefore the defect aware exploration and its novel objective function with the input from defect detection. For the experiments in the article a simulation of a water ballast tank was created based on 3D laser scans from a real double hull vessel. The results of the experiments showed that the UAVs were able to explore and map the ballast tank and construct a metric occupancy grid of the environment, while at the same time detecting defects along the path.

Article G

Towards an Autonomous, Visual Inspection-aware 3D Exploration and Mapping System for Water Ballast Tanks of Marine Vessels

Rune Y. Brogaard[1,*], Rasmus Eckholdt Andersen[1] , Luka Kovac[1], Marcin Zajaczkowski[2], Evangelos Boukas[1]

[1]Department of Electrical and Photonics Engineering, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

[2]FORCE Technology, Park Allé 345, Brøndby, Denmark

Abstract:

While there exist several approaches for autonomous exploration of confined spaces, when it comes to inspection tasks, there are not many works that take into account the actual classification of defects in the robot's view. Such approaches may result in less than optimal observations —even miss corrosion areas— and, therefore, can hinder the fulfillment of the purpose of the exploration, which is the identification of defects. In this paper, the first steps towards the unification of the exploration and inspection procedures, are theorized and experimentally evaluated. The next-best-view algorithm is augmented with information stemming from the defect classification. The system is tested within a computer model of a water ballast tank, from a double-sided cargo carrier, infused with corrosion areas. The initial results indicate that the UAV is able to successfully maneuver through tight entrance ways to explore and map the different compartments of a ballast tank while observing a higher percentage of corrosion than "vanilla" next best view.

G.1 Introduction

The seaworthiness of marine vessels is mainly determined by their structural integrity and overall condition. To assess this integrity, it is of utmost importance that a thorough visual inspection is performed on the vessels' key components. Such key components are, among others, the ballast tanks, where it is often difficult and dangerous for human inspection personnel to enter. Due to the hostile environment, surveyors tend to miss assessing key

areas within the ballast tanks. Using an Unmanned Aerial Vehicle (UAV) for the visual assessment, can ensure a repeatable and thorough inspection.

Typical autonomous robot exploration of unknown spaces aims at generating a full map of the areas while defining collision-free regions and trajectories. In more complex cases —such as the autonomous visual inspection of marine vessels— where exploration aims at providing additional data, the requirements for the motion planing are augmented. Naturally, a safe, autonomous, and reliable motion planning is required to navigate the dark and confined spaces. However, further constraints, such as angle and distance of observation, are implied, and therefore, the online information of the in-view defect must be integrated in the exploration algorithm. To accomplish this, an exploration approach is therefore investigated which aims at maximizing volume coverage within the ballast tanks, while prioritizing defect areas.

In a similar manner to the work presented in [182] a stereo camera is used to generate local 3D maps in the robot's view. The sequential 3D maps are combined to form an occupancy map. By the end of the exploration, the whole mapped area is described as either occupied or free leafs of an octree. Iteratively, a random tree [184] is formed in the free space of the explored area. Each branch is scored based on the amount of unmapped space that can be mapped by following this branch and the amount of interest points (e.g. faults and defects in the case of inspection of ballast tanks) in the field of view of the on-board cameras. The detection of interest points is performed using custom trained deep learning algorithms [77]. The top scoring branch of the octree is selected, and the robot is commanded to move to the first node of this branch. The process is then repeated. The approach is tested in two simulated environments of a top-side water ballast tank (WBT) from a double-sided cargo vessel. The purpose of the first simulation is to show that the exploration planner is capable of mapping an entire ballast tank. The second simulation provides more photorealistic imagery, which will be used to demonstrate how a vision-based detection system can be used to detect points of interest that is then included in the scoring of each branch in the random tree.

The on-board calculation of paths to safely navigate and explore the unknown space, while properly observing possible defect areas, enables efficient and accurate inspection of the ballast tanks.

G.2 Related Work

Using UAVs for path planning for industrial inspections has been investigated by [186]. However, they assume a fully known environment and do not consider unknown changes to the structure/map. This assumption of a valid map/drawing is rarely true for marine vessels, as they often undergo minor structural improvements and repair throughout their lifespan. These minor structural changes are seldom added to the vessel drawings. Therefore, this paper assumes that we have no prior knowledge of the confined space of the ballast tanks. The authors in [182] present an algorithm for exploration of unknown and limited confined spaces. This algorithm yields good results for objects with a significant free space, either within or surrounding the object. However, it also provides results only for a single confined space and not a combination of multiple spaces, such as the compartments of a water

ballast tank. The multiple compartment issue, complicates the volume that needs to be explored and will therefore also be addressed in this paper. The detection of corrosion in ballast tanks is crucial for the classification of the vessels as well as the maintenance and general safety of the vessel, and therefore the crew onboard. Typically, autonomous defect detection was based on image processing techniques employing hand-crafted, empirical rules. The rise of deep learning over the past decade allows the exploitation of alternative (typically higher order) feature spaces [187] which provide a more descriptive representation of the environment [188]. Different approaches for the detection of corrosion have been described in [77]. On the one hand, the detection of interest areas (object detection) was described as a means of an overall assessment. FasterRCNN is a highly acknowledged region proposal and classification network with application areas both in the inspection of known environments [77] and in the exploration of novel regions [189]. In [77] Faster-RCNN —an object detector— and its derivative Mask-RCNN —and image segmentation network— have been investigated in the context of inspection. Even though the overall accuracy of the Mask-RCNN is higher, the speed of Faster-RCNN allows for onboard implementation inside our modified version of the next best view. In [92] regression of the percentage of corrosion in an image. The integration of this approach in an exploration methods is interesting, but was deemed out of scope for this first attempt at an inspection-aware exploration.

G.3 System Overview

G.3.1 Aerial Vehicles

The aerial platforms used in this paper are based on their availability in the simulation software and small size, which allows them to go through the narrow passages between the compartments in the ballast tank. For the Gazebo simulation, we use an Astec Firefly, shown in figure G.1, and in Airsim we use a DJI F450 frame-based UAV.

Figure G.1. The Unmanned Aerial Vehicle used for mapping an exploration inside simulated model of a water ballast tank of a marine vessel.

They utilize publicly available components, with a focus on minimizing size and weight. They are equipped with a Flight Controller Unit (FCU), and a small onboard Linux pc. To sense the surroundings, the UAVs are equipped with a stereo vision camera, which includes a time-synchronized IMU. The camera is equipped with an IR-projector, which aids in generating a point cloud used for mapping the environment. This camera setup makes it possible to both have a VIO system, in our case ROVIO [25], running while at the same time generating point clouds for detecting 3D features in the environment. Although it is known that viusal odometry produces drift [122, 131] and that absolute localization approaches could assist in eliminating the error [3, 144, 145, 4], this is deemed to be out of the scope and to be included in our future work.

It is assumed that the UAVs carry their own light source to illuminate the dark environment of the ballast tank. Due to computational limitations on the simulation hardware, this is only included in the second UAV model, but an UAV-based light source has proven to work both for VIO and point cloud generation for real-world tests [2]. Therefore, general artificial low-light conditions are modeled for the inside of the water ballast tank.

G.3.2 Environment

The used UAV-based systems aims to operate in the confined spaces of empty water ballast tanks on-board cargo vessels. Most of the time, the water ballast tanks are empty due to cargo being present in the cargo holds. It is therefore reasonable to assume that an airborne UAV can be utilized more often than a submerged vehicle.

To allow rapid development and testing —and due to temporary access restrictions to vessels (Covid-19)— a computer model of a topside water ballast tank of a double-sided cargo carrier was designed for the simulation software in both Gazebo[190] and AirSim [191]. The model of the water ballast tank is based on a 3D laser scan from a real vessel, and the comparison between the scan and the model can be seen in figure G.2.

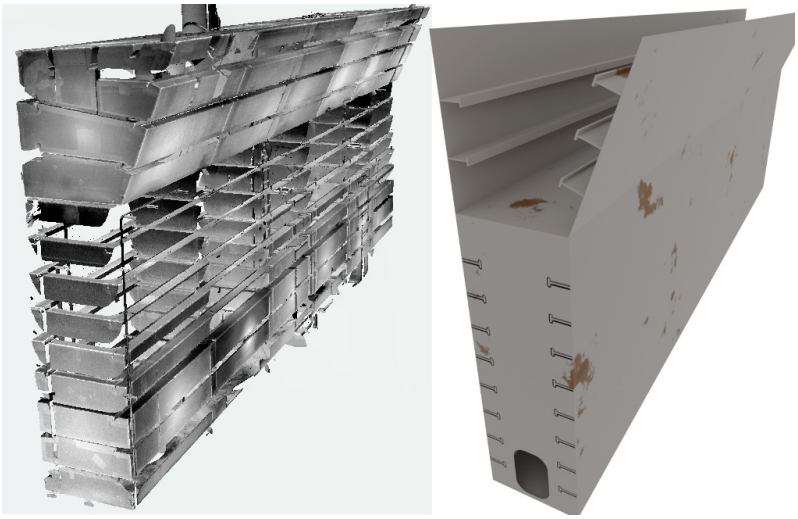


Figure G.2. An illustration of a typical water ballast tank from the port side of a double sided cargo carrier vessel. The point cloud (left) was captured using a FARO laser scanner with accuracy greater than 1 mm. The computer model (right) was build based on the point cloud from the real tank, and includes manholes, pipes, ect. as shown in figure G.3

The entire ballast tank is 18 meters long (x), 2 meters wide (y), 9 meters high (z), and consist of 2 levels. The top level is 3 meters tall and is divided into 3 sections. The bottom section is divided into five sections, each 3.5 meters long. A 2D view with separations (black lines) for each compartment of the ballast tank can be seen in figure G.8. It is possible to access the different sections through entrance holes between the sections near the floor of the section. Transitions between the two levels are only possible in compartment E and H, as shown in figure G.8.

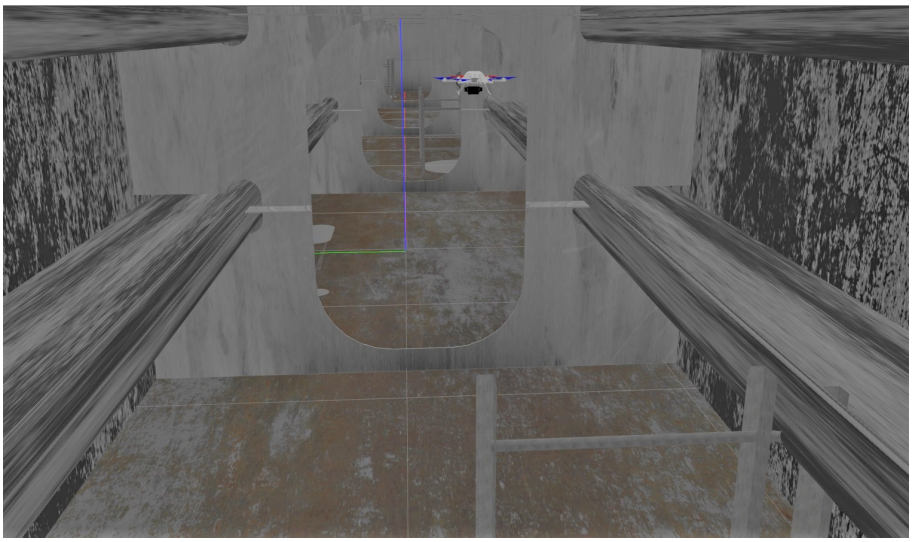


Figure G.3. The inside of the water ballast tank shown in figure G.2. The UAV in the top center of the image is located at (0,0.5,1) in the tank coordinate system, and can be seen through a passageway between two compartments in the bottom of the tank structure.

G.3.3 Mapping

To be able to plan a 3D trajectory in the unknown environment of the ballast tanks, the UAVs need to gain volumetric information of its closest environment. This is done using the point cloud data from the stereo camera, to create a octree-based map [183] of the environment. The occupancy map is a grid that covers the whole volume. To create the grid, the volume is split into octants. An octant is a recursive subdivision of the hierarchical space. Every octant is divided into eight smaller octants, and so forth. The tree representation of this map is called an octree. Taking into account n, the depth (or resolution of the octree), the octomap comprises $2^n \times 2^n \times 2^n$ octants. The 3D volume to be explored is initially considered unknown. The leafs of the octree are either occupied, free, or unmapped. The octrees are selected in octomap for their simplicity, efficiency and functionality (e:g: traversing and closest child querry). In figure G.4 it can be seen how a map is gradually built up over a period of 30 minutes, while the UAV explores the water ballast tank.

G.4 Exploration and path planning

The goal of the planner is to maximize exploration while penalizing the travel distance and rewarding the observation of new defects. The foundation of the exploration and path planning is based on the work of the authors in [182], so the nomenclature in this section

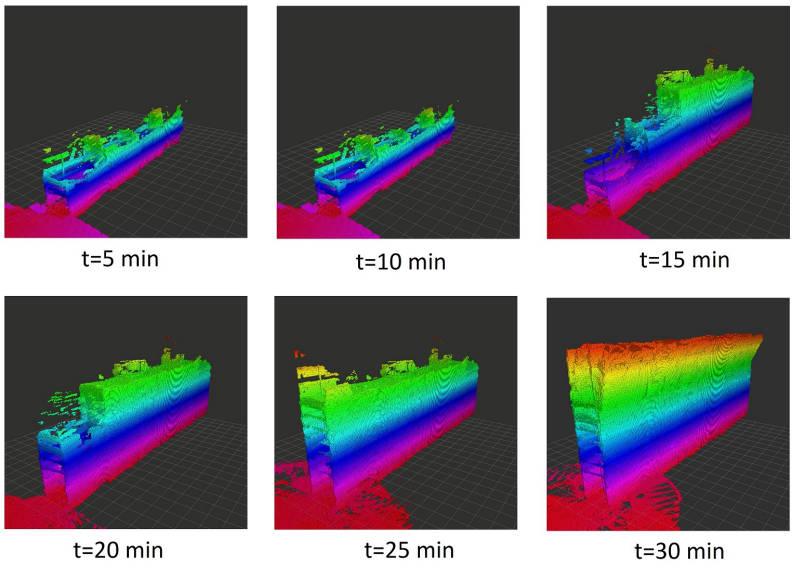


Figure G.4. Example of how a map is slowly being build, over a period of 30 minutes, as the UAV explores the water ballast tank from the inside.

follows that work closely. At each iteration of the path planning algorithm, only free and occupied space —in the octree— is considered. As a first step, RRT samples points in the free space of the octree. Then, samples (nodes of the RRT) are connected with a straight line if the connection volume is marked as free. RRT is chosen on the basis of its natural ability to handle obstacles and tight spaces. RRT* [192] would most likely give better results but would require more computational resources. An alternative to RRT could be Probabilistic Road Maps (PRM) [193], but it would spend more computational resources on collision checking when building edges to the surrounding nodes. RRT is, therefore, chosen for this paper. The state of the robot is given by $\xi = (x, y, z, \theta)$, which corresponds to the UAV being able to translate in 3D and rotate around the z-axis (yaw). The lack of roll and pitch in the state is explained by the drone flight stability and the slow straight line movements generated by the RRT based planner.

G.4.1 Original NBV Path Score and Selection

A path is given by $\sigma : \mathbb{R} \rightarrow \xi$, which corresponds to the path between two discrete vehicle configurations ξ_{k-1} and ξ_k. The paths between these configurations are checked for collision by diluting the path/edge in the shape of a box using the size of the drone as the size parameter. The path cost is defined as $c(\sigma^k_{k-1})$, and due to moving in straight lines, the cost is considered to be the Euclidean distance:

$$c(\sigma^k_{k-1}) = ||\mathbf{x}_k - \mathbf{x}_{k-1}||_2 \tag{G.1}$$

The yaw angle, θ, of the UAV is not considered in the cost. The maximum speed, v_{max}, and the maximum yaw angle, θ_{max}, of the UAV are controlled by the flight controller, which in our case is simulated by the RotorS package [194]. For each vehicle configuration, ξ, there is a certain set of voxels, m, which are both unmapped and can be explored/seen by the camera sensor. This set will be denoted as $E(M, \xi)$, and lies in the unexplored space V_{unmap}, and is important for the determining the reward for each node in the RRT. The quality/reward of each node $Q(n)$ in the tree is the sum of the unmapped volume that

can be explored at the nodes along the branch. For node k, this can be expressed as:

$$Q(n_k) = Q(n_{k-1}) + E(M, \xi) \cdot e^{-\lambda c(\sigma_{k-1}^k)} \tag{G.2}$$

Note that a tuning factor, λ, in equation G.2, penalizes large paths.

Starting from the current vehicle configuration, a tree is grown using RRT in the free space, and the node with the best reward is determined. The first node in that branch is set as the new destination for the UAV. A new tree can then be grown from the new position, but to prevent discarding high-quality paths already found, the previous branch with the best node is used to reinitialize the new tree. This principle is illustrated in figure G.5, where the black bold line indicates the best branch in the first tree, which the second tree is then built upon. The best node in any given tree is the one with the highest value. From the figure G.5 it can be seen that for the first tree the best node has a value of 193 and for the second tree the value is 187. To facilitate the interpretation of the figure, some of the nodes with smaller values are not displayed.

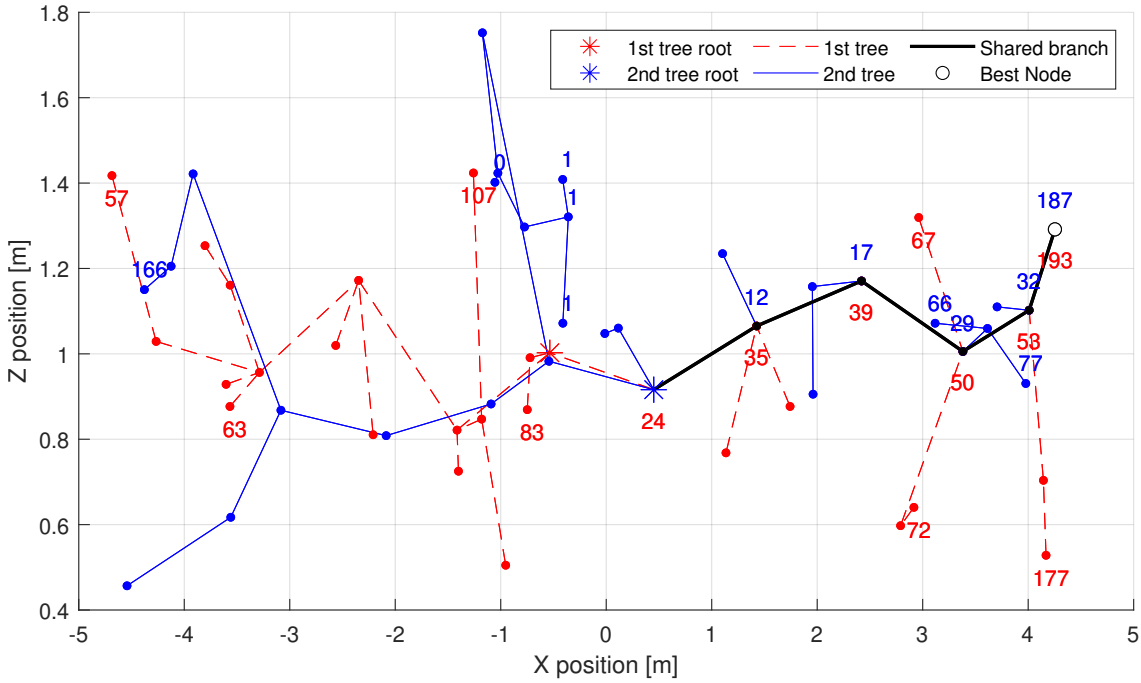


Figure G.5. Illustration of tree iterations, where the best branch from the first tree is used as the base for the second tree.

G.4.2 Novel Path Score

For inspection purposes, the goal is to spend as much time as possible looking at points of interest in the ballast tank, such as faults and defects. To incorporate this behavior, a deep learning model [77] is included on the drone to process the camera sensor information and map interest points in the 3D space. The algorithm is a Faster-RCNN network pre-trained on imagenet on which transfer learning was applied using a custom recorded dataset. For the implementation details the reader is directed to [77]. Equation G.2 is updated using a scaling factor μ based on how many interest points that are within the field of view in nodes:

$$Q(n_k) = Q(n_{k-1}) + \mu E(M, \xi) \cdot e^{-\lambda c(\sigma_{k-1}^k)} \tag{G.3}$$

The scaling factor is computed based on the sum of inverse distances:

$$\mu = \sum_{i \in I} \frac{1}{d_i} \tag{G.4}$$

Here I is the set of visible interest points in the camera FOV and d_i is the distance between interest point i and the camera. When the detection model detects a new interest point, the stereo camera is used to project the 2D image point to 3D and is stored with respect to a fixed frame. This means that when the exploration tree is expanded, the FOV can be computed for each node in the tree, and the amount of interest points within each FOV can be used to compute the sum of inverse distances to scale the quality/reward of a node.

G.5 Experiments

In order to evaluate the performance of the exploration system, a simulated environment is set up using ROS, Gazebo and AirSim using Unreal Engine. The unknown environment is represented by a computer model of a water ballast tank, as shown in figure G.2. The UAV has to explore this space by moving around inside the compartments. The UAV is located at the floor in the bottom level of the WBT in the middle compartment, with its vehicle configuration as $(x, y, z, \theta) = (0, 0, 0, 0)$. From this position the UAV is launched to 1 meter above the floor and the exploration algorithm is started. The parameters for the exploration algorithm in the Gazebo simulation can be seen in table G.1. To limit computational requirements, the RRT size is limited to N_T^{max} nodes, and the maximum length of each edge is set to 2 meters. To properly calculate the reward for each node, using equation G.2, the field of view of the camera is configured and the max depth field is set to 5 meters. Gaussian noise is furthermore added to the sensors, including the camera, to closely resemble real-world conditions. Previous work in [77] investigated the use of deep

Table G.1. Parameters for exploring the Water Ballast Tank in Gazebo

Parameter	**Value**	**Parameter**	**Value**
v_{max}	0.25 m/s	Volume (x,y,z)	16x2x9m
θ_{max}	0.3 rad/s	Map resolution	0.1m
λ	0.2	N_T^{max}	30

learning architectures for inspection in marine vessel ballast tanks. The AirSim simulation is included because of the ability to generate more photo-realistic environments that is required when running a deep learning model for interest point detection. The detection of interest points in this paper is done using a Faster-RCNN model trained on images generated in Blender [185], although it is independent of the exploration system and could be any detection model. An example of highlighted interest points is shown in figure G.6. The center of the bounding boxes is used to look up the 3D coordinate representing each point of interest. The 3D coordinate is then used to scale the gain by equation G.3. The exploration parameters for the AirSim simulation are listed in table G.2.

G.5.1 Results

The UAV in the Gazebo simulation is set to explore the WBT and an occupancy map is built by the UAV as shown in figure G.7. From this it can be seen that the UAV is able to

Table G.2. Parameters for exploring the Water Ballast Tank in AirSim

Parameter	**Value**	**Parameter**	**Value**
v_{max}	0.25 m/s	Volume (x,y,z)	16x2x9m
θ_{max}	0.3 rad/s	Map resolution	0.05m
λ	1.0	N_T^{max}	180

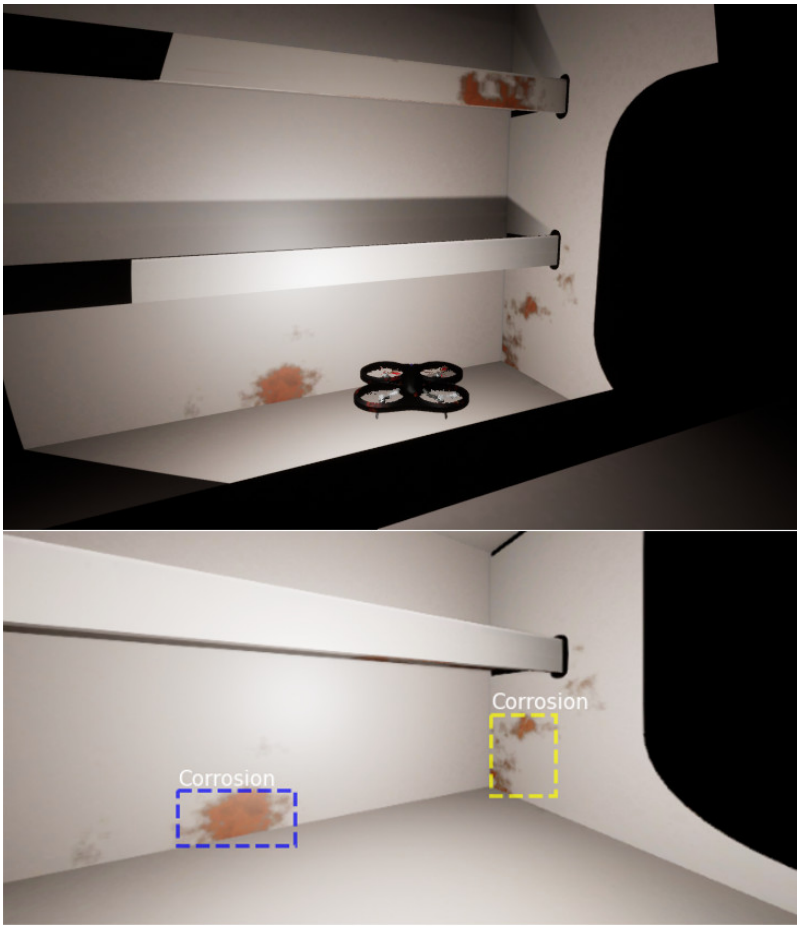


Figure G.6. (top) The AirSim drone located in front of a corrosion (bottom) Example output of the detection model detecting corrosion in the image from the camera on the drone.

map the WBT with the exception of some lack of data for the front center of the tank. The actual path of the drone during the exploration is shown in figure G.8 as the blue line. The white and black dots indicate the start and end positions, respectively. Due to the relatively flat structure of the WBT, the path is depicted as the projection from 3D to 2D, so that the y-axis is discarded. This eases the interpretation and results in a sideways view of the ballast tank, where the compartments are marked by the thick black lines, and given an identification letter from A to H in the top left corner. For every planning step, a tree with 30 nodes is being built. The complete set of the 460 trees is shown as the red dots (nodes) and the gray lines between the nodes (edges).

It can be seen from the path that the UAV crosses the different compartments through the entrance ways in the bottom part of the WBT. Connections between the top layer (A to C) and the bottom layer (E to H) are only available at the tops of D and H. However, the UAV still passes through the top of compartment G. This is caused by errors in the

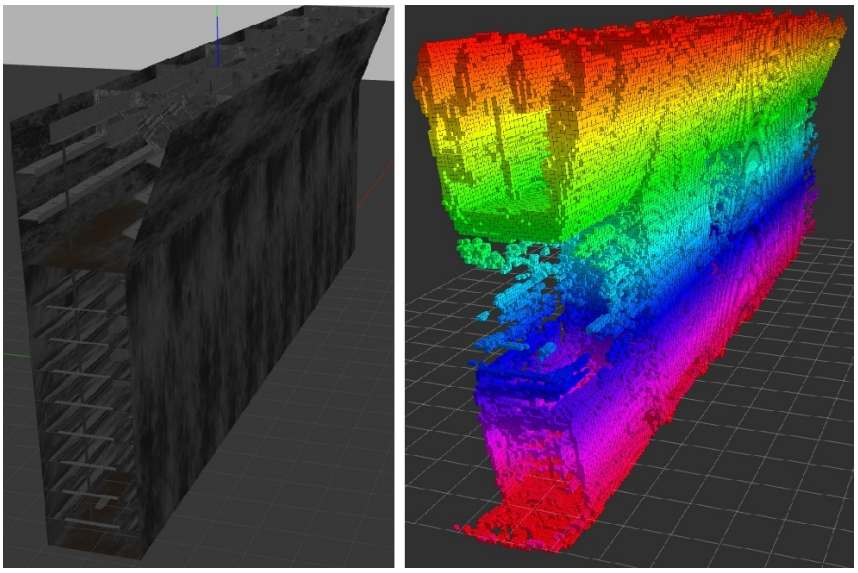


Figure G.7. The original model of the water ballast tank (left) compared to the explored and mapped model (right) represented as an occupancy grid map. The colour depicts the height of each voxel.

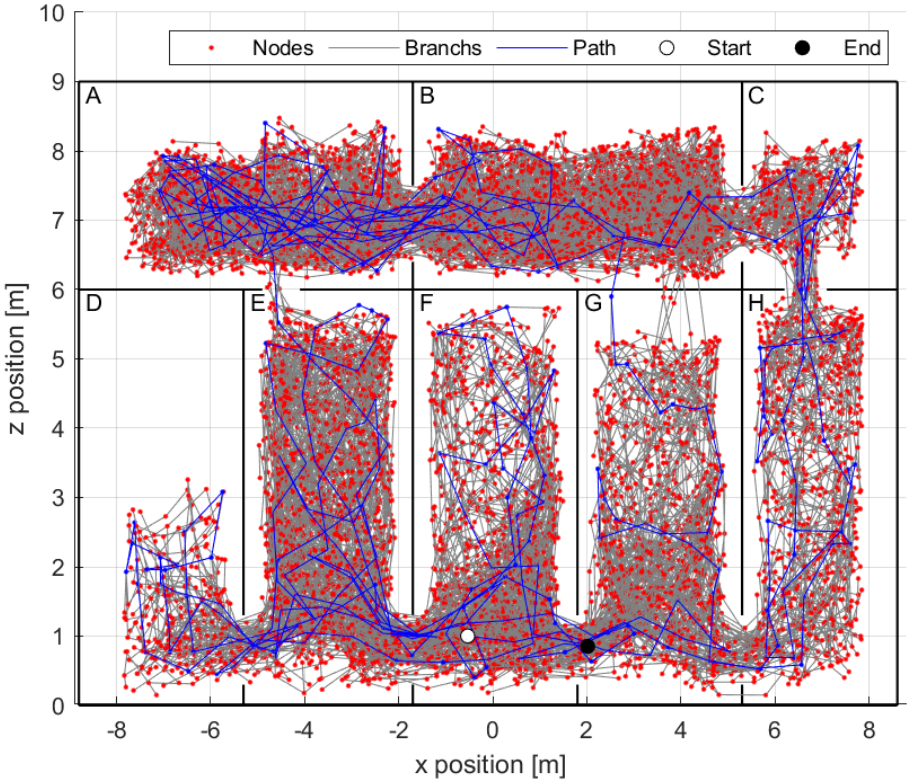


Figure G.8. Illustration of the 460 generated exploration trees and its nodes projected to 2D on the side of the ballast tank. The blue line depicts the resulting x,z-path of the UAV.

simulation, where the roof of G can become transparent. In compartment E there is an obstruction that prevents the UAV from fully exploring the compartment.

To validate that the addition of interest points to the gain computation in equation G.3, the detection model is included in the AirSim simulation environment. The results of including the interest points in the gain computation is visible by the drone valuing locations that overlook more interest points. In our experiments, the number of detections increased from an average of 1.21 detections per image to 1.35 detections per image. This 10% difference directly translates into more images available for future repairmen's decisions or other subsequent planning based on the results of the inspection. The results of this first attempt at integrating defect detection into the exploration indicate the potential of the idea but, nonetheless, warrant further development and evaluation.

G.6 Conclusions

Within this work, a simple and computationally light system for exploring and mapping unknown confined spaces —while increasing defect observation— was investigated. The system was tested in two realistic simulated environments of a marine vessel water ballast tank, using an UAVs with stereo vision cameras. The UAVs were able to explore and map the ballast tank and construct a metric occupancy grid of the environment. Additionally, a modified gain computation was introduced that indicated an increase in drone positions with a higher number of interest points within the FOV. The interest points were found using an off-the-shelf deep learning Faster R-CNN architecture trained on images generated using Blender software. The detected interest points are projected to 3D space using the stereo camera on the drone, which allows for computing the sum of inverse distances scaling used in the modified scoring of the branches in the exploration tree.

The exploration system will in later iterations be improved by adding simple prior knowledge of the environment. Such knowledge for the ballast tanks could be the number of levels, compartments, and possible transition connections between compartments. These improvements are aimed at being implemented into a RRT* or a kind of informed RRT.

Chapter 7

Conclusion & Future Research

Inspection of industrial confined spaces poses serious risks to humans, and with a focus on the maritime industry, the purpose of this dissertation was to create an absolute localization system for an autonomous robotic inspection solution. A literature review showed that absolute localization within the confined spaces of marine vessels is still limited and does not exist for the water ballast tanks. The investigation also showed that wheel-based robotic platforms, such as crawlers, can have difficulty traversing confined spaces of vessels. Therefore, this work focused on the absolute localization aspect of both an aerial and legged-based robotic inspection solution. Different approaches were investigated, where the first method used custom structural landmarks extracted from an RGB-D camera, to update an absolute pose estimate. The landmarks were extracted by detecting known structural elements from the ballast tank of marine vessels and comparing them to the actual position in a map of the tank. Using this method, the UAV was able to accurately localize itself as well as artificial defects scattered around the tank. However, it was limited to a very specific environment due to the definition of custom landmarks. To mitigate this limitation, the next work introduced 3D feature descriptors to extract structural information from point clouds generated by the RGB-D camera. State-of-the-art deep learning-based feature descriptors were used to register a sensor scan to a known map of the environment. By fusing a Visual Inertial Odometry estimate with the absolute pose estimate from the 3D registration, the system was able to provide pose estimates without the limitations of the custom landmark approach. However, the system lacked the ability to provide the uncertainty of the 3D registration, which is required for proper state estimation. Therefore, the next work introduced a GPU-based uncertainty estimation method. The uncertainty approach was experimentally tested for both a chemical distillation column and a water ballast tank. The last article removed the assumption of available CAD drawings, where research was instead conducted on how to perform an autonomous inspection without a prior map of the environment. An exploration-based approach was used to explore the unknown confined space, with additional weight in the objective function for the collection of inspection data along the robot flight path.

As the reader may recall, the first objective of the PhD, described in Section 1.2, was to provide an absolute localization system for known confined spaces that was tailored to the data availability. This has been achieved by creating localization pipelines that could work with just a few custom 3D landmarks extracted from 2D drawings, as well as a system that could localize itself using a prior map from CAD drawings. Additionally, an

exploration-based inspection system was provided for unknown confined spaces without a prior map. The second objective of the Ph.D. was to provide an uncertainty estimate for a localization pipeline in visually degraded or featureless environments. This objective was achieved for a chemical distillation column with very limited structural and visual features. Similarly, an uncertainty estimate was implemented in one of the localization pipelines provided in the first objective, creating additional value for the resulting absolute pose estimation. The third and last objective was to ensure that an absolute localization pipeline could be created for online operation on small form-factor robots. A range of the discussed pipelines achieved this by utilizing a small Graphic Processing Unit, and were demonstrated in both the distillation column and the water ballast tank mock-up model. The work carried out during this project has shown that absolute localization can be obtained for industrial confined spaces using a variety of different solutions, and therefore I consider our objectives to have been fully achieved.

7.1 Future Research

Accurate localization for inspection of industrial confined spaces is still a novel field that leads to even more research directions. The findings of **article A** support this claim, where it was also found that the shipping industry did not have a fully automatic inspection solution for confined spaces. The usage of an end-to-end deep learning localization pipeline would be a highly interesting area, as the availability of smaller and more powerful GPUs has steadily increased in recent years. Several approaches could be used, where a 2D to 3D matching pipeline, using the Depth images for a registration to a CAD model, could be an area worth researching. This could simplify the pipeline and have the potential to reduce computational requirements compared to dense 3D feature extraction. Similarly, incorporating ML-based prediction of the most common areas of corrosion and buckling could increase the efficiency of an automated inspection process. Corrosion in steel assets additionally requires thickness measurements of the remaining material, and a contact-based navigation solution with ultra sonic thickness measurement capabilities would also add additional value to an automated inspection solution. Investigating the continued use of legged robots for inspecting confined spaces could, furthermore, help overcome some of the limitations inherited by the use of small unmanned aerial vehicles. As legged robots start to improve their stability and robustness in rough terrain, this could allow longer operation times and less issues with turbulence compared to aerial solutions. A legged robot could still provide some flexibility in traversing difficult-to-reach areas of the ship, but would have the possibility of larger sensor packs and compute power. By using larger spinning LIDARS this could for example increase the Field-of-View (FoV) compared to the very limited FoV of the sensor pack on the UAV used in this project. Incorporating air quality sensors onboard the robot could provide a valuable addition to a mapping system that maps not only the structure, but also air quality throughout the environment. This has the benefit of increasing the level of safety of human personnel during the repair of detected defects. Another large and obvious research area could also be an autonomous robotic system with repair capabilities. This would require surface preparation and, most likely, stable contact with the surface at the defect location. It is clear that robotic inspection for confined spaces still has a range of areas that require further research before a fully autonomous solution will be available for the industry.

References

[1] Rasmus E. Andersen, Rune Y. Brogaard, and Evangelos Boukas. Autonomous robotic inspection for Remote Inspection Technique Systems: A review. *Field Robotics*, 3:69–96, 2023. doi: https://doi.org/10.55417/fr.2023002.

[2] Rune Y. Brogaard, Marcin Zajaczkowski, Luka Kovac, Ole Ravn, and Evangelos Boukas. Towards UAV-Based Absolute Hierarchical Localization in Confined Spaces. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 182–188, 2020. doi: 10.1109/ssrr50563.2020.9292616.

[3] Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. Absolute localisation in confined spaces using deep geometric features. *Electronics Letters*, 57(16):621–623, 2021. doi: https://doi.org/10.1049/ell2.12199.

[4] Rune Y. Brogaard, Ole Ravn, and Evangelos Boukas. GPU-accelerated Localization in Confined Spaces using Deep Geometric Features. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6, 2021. doi: 10.1109/ist50367.2021.9651425.

[5] Rune Y. Brogaard and Evangelos Boukas. Autonomous GPU-Based UAS for Inspection of Confined Spaces: Application to marine vessel classification. *Robotics and Autonomous Systems (RAS)*, 2023.

[6] Rune Y. Brogaard, Robert A. Hewitt, Sarah Etter, Arash Kalantari, and Evangelos Boukas. Absolute Localization in Feature-poor Industrial Confined Spaces. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023.

[7] Rune Y. Brogaard, Rasmus E. Andersen, Luka Kovac, Marcin Zajaczkowski, and Evangelos Boukas. Towards an Autonomous, Visual Inspection-aware 3D Exploration and Mapping System for Water Ballast Tanks of Marine Vessels. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6. Ieee, 2021. doi: 10.1109/ist50367.2021.9651476.

[8] International Maritime Organization (IMO). Enclosed space fatalities aboard ships during the period 2000-2021. *AMENDMENTS TO THE IMSBC CODE AND SUPPLEMENTS*, 2021.

[9] U.S. Department of Labor. Confined Spaces. *Occupational Safety and Health Administration*, 2023.

[10] Jason Selman, Jeffrey Spickett, Janis Jansz, and Benjamin Mullins. An investigation into the rate and mechanism of incident of work-related confined space fatalities. *Safety science*, 109:333–343, 2018.

[11] US Bureau of Labor Statistics. Fatal occupational injuries involving confined spaces — July 2020. *Census of Fatal Occupational Injuries (CFOI)*, 2021.

[12] Alberto Ortiz, Kai Yao, Francisco Bonnin-Pascual, Emilio Garcia-fidalgo, and Joan P Company-corcoles. New Steps towards the Integration of Robotic and Autonomous Systems in the Inspection of Vessel Holds. *Jornadas Nacionales de Robótica (Spanish Robotics Workshop)*, 2018.

[13] Håkan Wirdelius and Elena Oesterberg. Study of defect characteristics essential for ndt testing methods et, ut and rt. *inis.iaea.org*, 2000.

[14] Janis Tiemann, Andrew Ramsey, and Christian Wietfeld. Enhanced UAV indoor navigation through SLAM-augmented UWB localization. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6. IEEE, 2018.

[15] Alwin Poulose and Dong Seog Han. UWB indoor localization using deep learning LSTM networks. *Applied Sciences*, 10(18):6290, 2020.

[16] Beiya Yang, Erfu Yang, Leijian Yu, and Andrew Loeliger. High-Precision UWB-Based Localisation for UAV in Extremely Confined Environments. *IEEE Sensors Journal*, 22(1):1020–1029, 2021.

[17] Luca Barbieri, Mattia Brambilla, Andrea Trabattoni, Stefano Mervic, and Monica Nicoli. UWB localization in a smart factory: Augmentation methods and experimental assessment. *IEEE Transactions on Instrumentation and Measurement*, 70:1–18, 2021.

[18] Nicola Macoir, Jan Bauwens, Bart Jooris, Ben Van Herbruggen, Jen Rossey, Jeroen Hoebeke, and Eli De Poorter. Uwb localization with battery-powered wireless backbone for drone-based inventory management. *Sensors*, 19(3):467, 2019.

[19] Sara Roos-Hoefgeest, Ignacio Alvarez Garcia, and Rafael C Gonzalez. Mobile robot localization in industrial environments using a ring of cameras and ArUco markers. In *IECON 2021–47th Annual Conference of the IEEE Industrial Electronics Society*, pages 1–6. IEEE, 2021.

[20] Chenjie Wang, Lu Yin, Qing Zhao, Wei Wang, Chengyuan Li, and Bin Luo. An intelligent robot for indoor substation inspection. *Industrial Robot: the international journal of robotics research and application*, 2020.

[21] Mingxuan Zhou and Fuqiang Zhou. An Active Vision Based Localization and Mapping Method for MAV Inspection in Visual-degraded Restricted Environment. In *2019 6th International Conference on Systems and Informatics (ICSAI)*, pages 133–138. IEEE, 2019.

[22] S. N. Sirimanne et al. A 3-dimensional sift descriptor and its application to action recognition. In *Review of maritime transport 2021*, pages 357–360, 2021.

[23] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 15–22. IEEE, 2014.

[24] Rui Wang, Martin Schworer, and Daniel Cremers. Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3903–3911, 2017.

[25] Michael Bloesch, Michael Burri, Sammy Omari, Marco Hutter, and Roland Siegwart. Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback. *The International Journal of Robotics Research*, 36(10): 1053–1072, 2017.

[26] T. Qin, P. Li, and S. Shen. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.

[27] SAE International. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles (SAE Standard J3016, Report No. J3016-202104), 2021.

[28] United Nations Conference on Trade and Development. *Review of Maritime Transport 2019*. United Nations Publications, 300 East 42nd Street, New York, United States of America, 2019. doi: https://doi.org/https://doi.org/10.18356/17932789-en. URL `https://www.un-ilibrary.org/content/publication/17932789-en`.

[29] International Maritime Organization. Introduction to IMO, 2015.

[30] International Association of Classification Societies. Rec 42 Guidelines for Use of Remote Inspection Techniques for surveys - Rev.2 June 2016, 2016.

[31] H. Zheng, L. X. Kong, and S. Nahavandi. Automatic inspection of metallic surface defects using genetic algorithms. *Journal of Materials Processing Technology*, 125-126:427–433, 2002. ISSN 09240136. doi: 10.1016/s0924-0136(02)00294-7.

[32] G Ji, Y Zhu, and Y Zhang. The corroded defect rating system of coating material based on computer vision. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7220 Lncs:210–220, 2012. doi: 10.1007/978-3-642-31439-1_19.

[33] Francisco Bonnin-Pascual and Alberto Ortiz. A probabilistic approach for defect detection based on saliency mechanisms. *19th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2014*, 2014. doi: 10.1109/etfa.2014.7005257.

[34] Lili Liu, Estee Tan, Xi Jiang Yin, Yongda Zhen, and Zhi Qiang Cai. Deep learning for Coating Condition Assessment with Active perception. In *Proceedings of the 2019 3rd High Performance Computing and Cluster Technologies Conference on - HPCCT 2019*, pages 75–80, New York, New York, USA, 2019. ACM Press. ISBN 9781450371858. doi: 10.1145/3341069.3342966.

[35] Paul Ozog and Ryan M. Eustice. Identifying structural anomalies in image reconstructions of underwater ship hulls. *OCEANS 2015 - MTS/IEEE Washington*, pages 1–7, February 2016. doi: 10.23919/oceans.2015.7404406.

[36] Franz S. Hover, Ryan M. Eustice, Ayoung Kim, Brendan Englot, Hordur Johannsson, Michael Kaess, and John J. Leonard. Advanced perception, navigation and planning for autonomous in-water ship hull inspection. *International Journal of Robotics Research*, 31(12):1445–1464, October 2012. ISSN 02783649. doi: 10.1177/0278364912461059.

[37] Alberto Ortiz, Francisco Bonnin-Pascual, Emilio Garcia-Fidalgo, and Joan P. Company-Corcoles. Vision-based corrosion detection assisted by a micro-aerial vehicle in a vessel inspection application. *Sensors (Switzerland)*, 16(12):1–29, 2016. ISSN 14248220. doi: 10.3390/s16122118.

[38] Francisco Bonnin-Pascual and Alberto Ortiz. On the use of robots and vision technologies for the inspection of vessels: A survey on recent advances, October 2019. ISSN 00298018.

[39] Lloyd's Register. Remote Inspection Technique Systems (RITS) Assessment Standard for use on LR Class Surveys of Steel Structure, 2018.

[40] Victoria J. Hodge, Simon O'Keefe, Michael Weeks, and Anthony Moulds. Wireless sensor networks for condition monitoring in the railway industry: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1088–1106, 2015. ISSN 15249050. doi: 10.1109/tits.2014.2366512.

[41] Mohammad R. Jahanshahi, Jonathan S. Kelly, Sami F. Masri, and Gaurav S. Sukhatme. A survey and evaluation of promising approaches for automatic image-based defect detection of bridge structures. *Structure and Infrastructure Engineering*, 5(6):455–486, 2009. ISSN 15732479. doi: 10.1080/15732470801945930.

[42] Markus Eich and Thomas Vögele. Design and control of a lightweight magnetic climbing robot for vessel inspection. *2011 19th Mediterranean Conference on Control and Automation, MED 2011*, 55(June):1200–1205, 2011. doi: 10.1109/med.2011.5983075.

[43] C. Balaguer, R. Montero, J. G. Victores, S. Martínez, and A. Jardón. Towards fully automated tunnel inspection: A survey and future trends. *31st International Symposium on Automation and Robotics in Construction and Mining, ISARC 2014 - Proceedings*, 31(January 2015):19–33, 2014. doi: 10.22260/isarc2014/0005.

[44] Timothy S. Newman. Survey of automated visual inspection. *Computer Vision and Image Understanding*, 61(2):231–262, 1995. ISSN 10773142. doi: 10.1006/cviu.1995.1017.

[45] Luc Vincent, Luc Vincent, and Pierre Soille. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991. ISSN 01628828. doi: 10.1109/34.87344.

[46] John Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Pami-8(6):679–698, 1986. ISSN 01628828. doi: 10.1109/tpami.1986.4767851.

[47] B.B. Zaidan, A.A. Zaidan, o. Hamdan Alanzani, and Rami Alnaqeib. Towards corrosion detection system. *International Journal of Computer Science*, 7(3):33–36, 2010. ISSN 1694-0814.

[48] Francisco Bonnin-Pascual and Alberto Ortiz. A Flying Tool for Sensing Vessel Structure Defects Using Image Contrast-Based Saliency. *IEEE Sensors Journal*, 16 (15):6114–6121, 2016. ISSN 1530437x. doi: 10.1109/jsen.2016.2578360.

[49] Francisco Bonnin-Pascual and Alberto Ortiz. A Generic Framework for Defect Detection on Vessel Structures based on Image Saliency. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–4. Ieee, 2016. ISBN 9781509013142. doi: 10.1109/etfa.2016.7733668.

[50] Francisco Bonnin-Pascual and Alberto Ortiz. A novel approach for defect detection on vessel structures using saliency-related features. *Ocean Engineering*, 149(August 2017):397–408, 2018. ISSN 00298018. doi: 10.1016/j.oceaneng.2017.08.024.

[51] Francisco Bonnin-Pascual and Alberto Ortiz. A saliency-boosted corrosion detector for the visual inspection of vessels. *Frontiers in Artificial Intelligence and Applications*, 300:176–185, 2017. ISSN 09226389. doi: 10.3233/978-1-61499-806-8-176.

[52] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8): 861–874, 2006. ISSN 01678655. doi: 10.1016/j.patrec.2005.10.010.

[53] Rosalia Maglietta, Annalisa Milella, Massimo Caccia, and Gabriele Bruzzone. A vision-based system for robotic inspection of marine vessels. *Signal, Image and Video Processing*, 12(3):471–478, 2018. ISSN 18631711. doi: 10.1007/s11760-017-1181-9.

[54] Francisco Bonnin-Pascual and Alberto Ortiz. An AdaBoost-based approach for coating breakdown detection in metallic surfaces. *2011 19th Mediterranean Conference on Control and Automation, MED 2011*, pages 1206–1211, 2011. doi: 10.1109/med.2011.5983121.

[55] Markus Eich, Francisco Bonnin-Pascual, Emilio Garcia-Fidalgo, Alberto Ortiz, Gabriele Bruzzone, Yannis Koveos, and Frank Kirchner. A robot application for marine vessel inspection. *Journal of Field Robotics*, 31(2):319–341, March 2014. ISSN 15564959. doi: 10.1002/rob.21498.

[56] Alberto Ortiz, Francisco Bonnin-Pascual, Emilio Garcia-Fidalgo, and Joan P. Company. Visual inspection of vessels by means of a micro-aerial vehicle: An artificial neural network approach for corrosion detection, 2016. ISSN 21945357.

[57] Michael T. Orchard and Charles A. Bouman. Color quantization of images. *IEEE transactions on signal processing*, 39(12):2677–2690, 1991. ISSN 16113349. doi: 10.1007/11582267_57.

[58] S Theodoridis and K Koutroumbas. Pattern Recognition, Passage. *Pattern Recognition, XP-002405837, San Diego, CA: Academic Press*, pages 238–241, 1999.

[59] David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035, 2007. ISSN 0547051x.

[60] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59, 1996. ISSN 00313203. doi: 10.1016/0031-3203(95)00067-4.

[61] Mel Siegel, Priyan Gunatilake, and Gregg Podnar. Robotic assistants for aircraft inspectors. *IEEE Instrumentation Measurement Magazine*, 1(March 1998):16–30, 1988.

[62] Mel Siegel and Priyan Gunatilake. Remote Enhanced Visual Inspection of Aircraft by a Mobile Robot. *Workshop on Emerging Technologies, Intelligent Measurement and Vistual Systems for Instrumentation and Measurement*, pages 1–10, 1998.

[63] Carlos Fernández-Isla, Pedro J. Navarro, and Pedro María Alcover. Automated visual inspection of ship hull surfaces using the wavelet transform. *Mathematical Problems in Engineering*, 2013, 2013. ISSN 1024123x. doi: 10.1155/2013/101837.

[64] Pedro Navarro, Andrés Iborra, Carlos Fernández, Pedro Sánchez, and Juan Suardíaz. A sensor system for detection of hull surface defects. *Sensors*, 10(8):7067–7081, 2010. ISSN 14248220. doi: 10.3390/s100807067.

[65] Pedro Javier Navarro, Juan Suardíaz Muro, Pedro María Alcover, and Carlos Fernández-Isla. Sensors systems for the automation of operations in the ship repair industry. *Sensors (Switzerland)*, 13(9):12345–12374, 2013. ISSN 14248220. doi: 10.3390/s130912345.

[66] Laurent Busin, Nicolas Vandenbroucke, and Ludovic Macaire. Color Spaces and Image Segmentation. *Advances in Imaging and Electron Physics*, 151(07):65–168, 2009. ISSN 10765670. doi: 10.1016/s1076-5670(07)00402-8.

[67] K. Y. Choi and S. S. Kim. Morphological analysis and classification of types of surface corrosion damage by digital image processing. *Corrosion Science*, 47(1):1–15, 2005. ISSN 0010938x. doi: 10.1016/j.corsci.2004.05.007.

[68] A. K. Aijazi, L. Malaterre, M. L. Tazir, L. Trassoudaine, and P. Checchin. Detecting and Analyzing Corrosion Spots on the Hull of Large Marine vessels using Colored 3D LIDAR Point Clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3(August 2018):153–160, 2016. ISSN 21949050. doi: 10.5194/isprs-annals-III-3-153-2016.

[69] Luca Petricca, Tomas Moss, Gonzalo Figueroa, and Stian Broen. Corrosion Detection Using A.I : A Comparison of Standard Computer Vision Techniques and Deep Learning Model. *International Conference on Computer Science, Engineering and Information Technology*, 91:91–99, 2016. doi: 10.5121/csit.2016.60608.

[70] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 Lncs:21–37, 2016. ISSN 16113349. doi: 10.1007/978-3-319-46448-0_2.

[71] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015. ISBN 0162-8828 Vo - Pp. doi: 10.1109/tpami.2016.2577031.

[72] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–14, 2014.

[73] Lili Liu, Estee Tan, Yongda Zhen, Xi Jiang Yin, and Zhi Qiang Cai. AI-facilitated coating corrosion assessment system for productivity enhancement. In *Proceedings of the 13th IEEE Conference on Industrial Electronics and Applications, ICIEA 2018*, pages 606–610. Institute of Electrical and Electronics Engineers Inc., June 2018. ISBN 9781538637579. doi: 10.1109/iciea.2018.8397787.

[74] Lili Liu, Estee Tan, Zhi Qiang Cai, Xi Jiang Yin, and Yongda Zhen. CNN-based automatic coating inspection system. *Advances in Science, Technology and Engineering Systems*, 3(6):469–478, 2018. ISSN 24156698. doi: 10.25046/aj030655.

[75] Lili Liu, Estee Tan, Zhi Qiang Cai, Yongda Zhen, and Xi Jiang Yin. An Integrated Coating Inspection System for Marine and Offshore Corrosion Management. In *2018 15th International Conference on Control, Automation, Robotics and Vision, ICARCV 2018*, pages 1531–1536. Institute of Electrical and Electronics Engineers Inc., December 2018. ISBN 9781538695821. doi: 10.1109/icarcv.2018.8581327.

[76] Blossom Treesa Bastian, Jaspreeth N, S. Kumar Ranjith, and C. V. Jiji. Visual inspection and characterization of external corrosion in pipelines using deep neural network. *NDT and E International*, 107(July):102134, October 2019. ISSN 09638695. doi: 10.1016/j.ndteint.2019.102134.

[77] Rasmus Andersen, Lazaros Nalpantidis, Ole Ravn, and Evangelos Boukas. Investigating Deep Learning Architectures towards Autonomous Inspection for Marine Classification. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 197–204. Ieee, 2020. doi: 10.1109/ssrr50563.2020.9292621.

[78] Kai Yao, Alberto Ortiz, and Francisco Bonnin-Pascual. A Weakly-Supervised Semantic Segmentation Approach Based on the Centroid Loss: Application to Quality Control and Inspection. *IEEE Access*, 9:69010–69026, 2021. ISSN 21693536. doi: 10.1109/access.2021.3077847.

[79] Ozan Oktay, Jo Schlemper, Loïc Le Folgoc, Matthew C. H. Lee, Mattias P. Heinrich, Kazunari Misawa, Kensaku Mori, Steven G. McDonagh, Nils Y. Hammerla,

Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention U-Net: Learning Where to Look for the Pancreas. *CoRR*, abs/1804.03999, 2018.

[80] S. Avril, A. Vautrin, and Y. Surrel. Grid method: Application to the characterization of cracks. *Experimental Mechanics*, 44(1):37–43, 2004. ISSN 00144851. doi: 10.1007/bf02427974.

[81] T. S. Akinfiev, M. A. Armada, and R. Fernandez. Nondestructive testing of the state of a ship's hull with an underwater robot. *Russian Journal of Nondestructive Testing*, 44(9):626–633, 2008. ISSN 10618309. doi: 10.1134/s1061830908090064.

[82] G. Dini and M. Dalle Mura. Application of Augmented Reality Techniques in Through-life Engineering Services. In *Procedia CIRP*, volume 38, pages 14–23. Elsevier B.V., January 2015. doi: 10.1016/j.procir.2015.07.044.

[83] Christos Papachristos and Kostas Alexis. Augmented reality-enhanced structural inspection using aerial robots. *IEEE International Symposium on Intelligent Control - Proceedings*, 2016-Septe:22–27, 2016. doi: 10.1109/isic.2016.7579983.

[84] C. P. Gardiner and R. E. Melchers. Corrosion analysis of bulk carriers, Part I: Operational parameters influencing corrosion rates. *Marine Structures*, 16(8): 547–566, 2003. ISSN 09518339. doi: 10.1016/s0951-8339(01)00026-0.

[85] N. Qaddoumi, A. Shroyer, and R. Zoughi. Microwave detection of rust under paint and composite laminates. *Research in Nondestructive Evaluation*, 9(4):201–212, 1997. ISSN 14322110. doi: 10.1080/09349849709414474.

[86] Mohammad R. Jahanshahi, Sami F. Masri, Curtis W. Padgett, and Gaurav S. Sukhatme. An innovative methodology for detection and quantification of cracks through incorporation of depth perception. *Machine Vision and Applications*, 24(2): 227–241, 2011. ISSN 09328092. doi: 10.1007/s00138-011-0394-0.

[87] Alberto Ortiz, Francisco Bonnin-Pascual, Emilio Garcia-Fidalgo, Joan P. Company-Corcoles, Francisco Bonnín-Pascual, Emilio García-Fidalgo, and Joan Pep Company-Córcoles. The INCASS Project Approach towards Automated Visual Inspection of Vessels. *Jornadas Nacionales de Robótica (Spanish Robotics Conference)*, 2017.

[88] Xincong Yang, Heng Li, Yantao Yu, Xiaochun Luo, Ting Huang, and Xu Yang. Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network. *Computer-Aided Civil and Infrastructure Engineering*, 33(12):1090–1109, 2018. ISSN 14678667. doi: 10.1111/mice.12412.

[89] Vedhus Hoskere, Yasutaka Narazaki, Tu A. Hoang, and B. F. Spencer. MaDnet: multi-task semantic segmentation of multiple types of structural materials and damage in images of civil infrastructure. *Journal of Civil Structural Health Monitoring*, 10(5):757–773, 2020. ISSN 21905479. doi: 10.1007/s13349-020-00409-0.

[90] Hyung Kyun Lim and Joo Sung Lee. On the structural behavior of ship's shell structures due to impact loading. *International Journal of Naval Architecture and*

Ocean Engineering, 10(1):103–118, jan 2018. ISSN 20926790. doi: 10.1016/j.ijnaoe.2017.03.002.

[91] Rasmus Eckholdt Andersen, Lazaros Nalpantidis, and Evangelos Boukas. Vessel Classification Using A Regression Neural Network Approach. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4480–4486. Ieee, 2021. doi: 10.1109/iros51168.2021.9636161.

[92] Rasmus Eckholdt Andersen, Lazaros Nalpantidis, Ole Ravn, and Evangelos Boukas. Simultaneous regression-based spatial coverage estimation and object detection with deep learning. *Electronics Letters*, 57(16):605–607, 2021. doi: https://doi.org/10.1049/ell2.12183.

[93] Thomas Lee, Susan McKeever, and Jane Courtney. Flying free: A research overview of deep learning in drone navigation autonomy. *Drones*, 5(2):1–18, 2021. ISSN 2504446x. doi: 10.3390/drones5020052.

[94] Tirthankar Bandyopadhyay, Ryan Steindl, Fletcher Talbot, Navinda Kottege, Ross Dungavell, Brett Wood, James Barker, Karsten Hoehn, and Alberto Elfes. Magneto: A Versatile Multi-Limbed Inspection Robot. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2253–2260. Institute of Electrical and Electronics Engineers Inc., December 2018. ISBN 9781538680940. doi: 10.1109/iros.2018.8593891.

[95] M. Caccia, R. Robino, W. Bateman, M. Eich, A. Ortiz, L. Drikos, A. Todorova, I. Gaviotis, F. Spadoni, and V. Apostolopoulou. *MINOAS a Marine INspection rObotic Assistant: System requirements and design*, volume 7. Ifac, 2010. ISBN 9783902661876. doi: 10.3182/20100906-3-it-2019.00083.

[96] Haocai Huang, Danhua Li, Zhao Xue, Xian Lei Chen, Shuyu Liu, Jianxing Leng, and Yan Wei. Design and performance analysis of a tracked wall-climbing robot for ship inspection in shipbuilding. *Ocean Engineering*, 131:224–230, February 2017. ISSN 00298018. doi: 10.1016/j.oceaneng.2017.01.003.

[97] Mohamed G. Alkalla, Mohamed A. Fanni, and Abdelfatah M. Mohamed. A novel propeller-type climbing robot for vessels inspection. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, volume 2015-Augus, pages 1623–1628. Institute of Electrical and Electronics Engineers Inc., August 2015. ISBN 9781467391078. doi: 10.1109/aim.2015.7222776.

[98] Mohamed G. Alkalla, Mohamed A. Fanni, and Abdel Fatah Mohamed. Versatile climbing robot for vessels inspection. In *Proceedings - 2015 International Conference on Control, Automation and Robotics, ICCAR 2015*, pages 18–23. Institute of Electrical and Electronics Engineers Inc., July 2015. ISBN 9781467375238. doi: 10.1109/iccar.2015.7165995.

[99] Emilio Garcia-Fidalgo, Alberto Ortiz, Francisco Bonnin-Pascual, and Joan P. Company. A mosaicing approach for vessel visual inspection using a micro-aerial vehicle. In *IEEE International Conference on Intelligent Robots and Systems*,

volume 2015-Decem, pages 104–110. Institute of Electrical and Electronics Engineers Inc., December 2015. ISBN 9781479999941. doi: 10.1109/iros.2015.7353361.

[100] Robert Schattschneider, Guillaume Maurino, and Wenhui Wang. Towards stereo vision SLAM based pose estimation for ship hull inspection. *OCEANS'11 - MTS/IEEE Kona, Program Book*, 2011. doi: 10.23919/oceans.2011.6106988.

[101] Dongha Chung and Jinwhan Kim. Pose Estimation Considering an Uncertainty Model of Stereo Vision for In-Water Ship Hull Inspection. *IFAC-PapersOnLine*, 51 (29):400–405, January 2018. ISSN 24058963. doi: 10.1016/j.ifacol.2018.09.454.

[102] Shahriar Negahdaripour and Pezham Firoozfam. An ROV stereovision system for ship-hull inspection. *IEEE Journal of Oceanic Engineering*, 31(3):551–564, 2006. ISSN 03649059. doi: 10.1109/joe.2005.851391.

[103] Annalisa Milella, Rosalia Maglietta, Massimo Caccia, and Gabriele Bruzzone. Robotic inspection of ship hull surfaces using a magnetic crawler and a monocular camera. *Sensor Review*, 37(4):425–435, 2017. ISSN 02602288. doi: 10.1108/sr-02-2017-0021.

[104] Luciano L. Menegaldo, Melquisedec Santos, Gustavo Andre Nunes Ferreira, Rodrigo Guerato Siqueira, and Lucas Moscato. SIRUS: A mobile robot for floating production storage and offloading (FPSO) ship hull inspection. In *International Workshop on Advanced Motion Control, AMC*, volume 1, pages 27–32, 2008. ISBN 9781424417032. doi: 10.1109/amc.2008.4516036.

[105] Paul Ozog, Matthew Johnson-Roberson, and Ryan M. Eustice. Mapping underwater ship hulls using a model-assisted bundle adjustment framework. *Robotics and Autonomous Systems*, 87:329–347, January 2017. ISSN 09218890. doi: 10.1016/j.robot.2016.09.006.

[106] Paul Ozog, Nicholas Carlevaris-Bianco, Ayoung Kim, and Ryan M. Eustice. Long-term Mapping Techniques for Ship Hull Inspection and Surveillance using an Autonomous Underwater Vehicle. *Journal of Field Robotics*, 33(3):265–289, May 2016. ISSN 15564967. doi: 10.1002/rob.21582.

[107] Ayoung Kim and Ryan Eustice. Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, pages 1559–1565. Ieee, December 2009. ISBN 9781424438044. doi: 10.1109/iros.2009.5354132.

[108] Geoffrey A. Hollinger, Brendan Englot, Franz Hover, Urbashi Mitra, and Gaurav S. Sukhatme. Uncertainty-driven view planning for underwater inspection. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4884–4891. Institute of Electrical and Electronics Engineers Inc., 2012. ISBN 9781467314039. doi: 10.1109/icra.2012.6224726.

[109] Brendan Englot and Franz Hover. Planning complex inspection tasks using redundant roadmaps. In *International Symposium on Robotics Research*, pages 327–343, 2011. ISBN 9783319293622. doi: 10.1007/978-3-319-29363-9_19.

[110] Raihan Enjikalayil Abdulkader, Prabakaran Veerajagadheswar, Nay Htet Lin, Selva Kumaran, Suresh Raj Vishaal, and Rajesh Elara Mohan. Sparrow: A magnetic climbing robot for autonomous thickness measurement in ship hull maintenance. *Journal of Marine Science and Engineering*, 8(6):469, June 2020. ISSN 20771312. doi: 10.3390/jmse8060469.

[111] Francisco Bonnin-Pascual, Emilio Garcia-Fidalgo, and Alberto Ortiz. Semi-autonomous visual inspection of vessels assisted by an unmanned Micro Aerial Vehicle. In *IEEE International Conference on Intelligent Robots and Systems*, pages 3955–3961, 2012. ISBN 9781467317375. doi: 10.1109/iros.2012.6385891.

[112] Francisco Bonnin-Pascual, Alberto Ortiz, Emilio Garcia-Fidalgo, and Joan P. Company. A Micro-Aerial platform for vessel visual inspection based on supervised autonomy. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2015-Decem, pages 46–52. Institute of Electrical and Electronics Engineers Inc., December 2015. ISBN 9781479999941. doi: 10.1109/iros.2015.7353353.

[113] Gordon Cheng and Alexander Zelinsky. Supervised autonomy: A framework for human-robot systems development. *Autonomous Robots*, 10(3):251–266, May 2001. ISSN 09295593. doi: 10.1023/a:1011231725361.

[114] Zheng Fang, Shichao Yang, Sezal Jain, Geetesh Dubey, Stephan Roth, Silvio Maeta, Stephen Nuske, Yu Zhang, and Sebastian Scherer. Robust Autonomous Flight in Constrained and Visually Degraded Shipboard Environments. *Journal of Field Robotics*, 34(1):25–52, 2017. ISSN 15564967. doi: 10.1002/rob.21670.

[115] Tung Dang, Shehryar Khattak, Frank Mascarich, and Kostas Alexis. Explore locally, plan globally: A path planning framework for autonomous robotic exploration in subterranean environments. In *2019 19th International Conference on Advanced Robotics, ICAR 2019*, pages 9–16. Institute of Electrical and Electronics Engineers Inc., December 2019. ISBN 9781728124674. doi: 10.1109/icar46387.2019.8981594.

[116] Tung Dang, Frank Mascarich, Shehryar Khattak, Huan Nguyen, Hai Nguyen, Satchel Hirsh, Russell Reinhart, Christos Papachristos, and Kostas Alexis. Autonomous Search for Underground Mine Rescue Using Aerial Robots. In *IEEE Aerospace Conference Proceedings*, pages 1–8. IEEE Computer Society, March 2020. ISBN 9781728127347. doi: 10.1109/aero47225.2020.9172804.

[117] Tung Dang, Marco Tranzatto, Shehryar Khattak, Frank Mascarich, Kostas Alexis, and Marco Hutter. Graph-based subterranean exploration path planning using aerial and legged robots. *Journal of Field Robotics*, 37(8):1363–1388, December 2020. ISSN 1556-4959. doi: 10.1002/rob.21993.

[118] Marco Hutter, Christian Gehring, Dominic Jud, Andreas Lauber, C. Dario Bellicoso, Vassilios Tsounis, Jemin Hwangbo, Karen Bodie, Peter Fankhauser, Michael Bloesch, Remo Diethelm, Samuel Bachmann, Amir Melzer, and Mark Hoepflinger. ANYmal - A highly mobile and dynamic quadrupedal robot. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2016-Novem, pages 38–44. Institute of Electrical and Electronics Engineers Inc., nov 2016. ISBN 9781509037629. doi: 10.1109/iros.2016.7758092.

[119] C. Gehring, P. Fankhauser, L. Isler, R. Diethelm, S. Bachmann, M. Potz, L. Gerstenberg, and M. Hutter. ANYmal in the Field: Solving Industrial Inspection of an Offshore HVDC Platform with a Quadrupedal Robot. In *Field and Service Robotics*, pages 247–260. Springer, Singapore, 2021. doi: 10.1007/978-981-15-9460-1_18.

[120] V. V. Kostenko, A. Yu Bykanova, and A. Yu Tolstonogov. Underwater Robotics Complex for Inspection and Laser Cleaning of Ships from Biofouling. In *IOP Conference Series: Earth and Environmental Science*, volume 272, page 022103. Institute of Physics Publishing, June 2019. doi: 10.1088/1755-1315/272/2/022103.

[121] Lloyd's Register Group. Rules and Regulationsfor the Classificationof Naval Ships. *Lloyd's Register Group*, 2020.

[122] Ioannis Kostavelis, Evangelos Boukas, Lazaros Nalpantidis, and Antonios Gasteratos. Stereo-based visual odometry for autonomous robot navigation. *International Journal of Advanced Robotic Systems*, 13(1):21, 2016.

[123] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart. Maplab: An Open Framework for Research in Visual-Inertial Mapping and Localization. *IEEE Robotics and Automation Letters*, 3(3):1418–1425, 2018.

[124] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Transactions on Robotics*, 33(5): 1255–1262, 2017.

[125] Loukas Bampis, Angelos Amanatiadis, and Antonios Gasteratos. Fast loop-closure detection using visual-word-vectors from image sequences. *The International Journal of Robotics Research*, 37(1):62–82, 2018.

[126] Angelos Amanatiadis. A multisensor indoor localization system for biped robots operating in industrial environments. *IEEE Transactions on Industrial Electronics*, 63(12):7597–7606, 2016.

[127] Huai Yu, Weikun Zhen, Wen Yang, Ji Zhang, and Sebastian Scherer. Monocular Camera Localization in Prior LiDAR Maps with 2D-3D Line Correspondences. In *Proceedings of (IROS) IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE/RSJ, 2020.

[128] Roberto Opromolla, Giancarmine Fasano, Giancarlo Rufino, Michele Grassi, and Al Savvaris. LIDAR-inertial integration for UAV localization and mapping in complex environments. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 649–656. IEEE, 2016.

[129] B. Kakillioglu, J. Wang, S. Velipasalar, A. Janani, and E. Koch. 3D Sensor-Based UAV Localization for Bridge Inspection. *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 1926–1930, 2019.

[130] J. Surber, L. Teixeira, and M. Chli. Robust visual-inertial localization with weak GPS priors for repetitive UAV flights. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6300–6306, 2017.

[131] George Lentaris, Ioannis Stamoulias, Dionysios Diamantopoulos, Konstantinos Maragos, Kostas Siozios, Dimitrios Soudris, Marcos Aviles Rodrigalvarez, Manolis Lourakis, Xenophon Zabulis, Ioannis Kostavelis, et al.
SPARTAN/SEXTANT/COMPASS: advancing space rover vision via reconfigurable platforms. In *International Symposium on Applied Reconfigurable Computing*, pages 475–486. Springer, 2015.

[132] Kostas Siozios, Dionysios Diamantopoulos, Ioannis Kostavelis, Evangelos Boukas, Lazaros Nalpantidis, Dimitrios Soudris, Antonios Gasteratos, Marcos Avilés, and Iraklis Anagnostopoulos. SPARTAN project: Efficient implementation of computer vision algorithms onto reconfigurable platform targeting to space applications. In *6th International Workshop on Reconfigurable Communication-Centric Systems-on-Chip (ReCoSoC)*, pages 1–9. Ieee, 2011.

[133] Lorenz Meier, Dominik Honegger, and Marc Pollefeys. PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 6235–6240. Ieee, 2015.

[134] Helmut Pottmann, Qi-Xing Huang, Yong-Liang Yang, and Shi-Min Hu. Geometry and convergence analysis of algorithms for registration of 3D shapes. *International Journal of Computer Vision*, 67:277–296, 2006.

[135] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (FPFH) for 3D registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009.

[136] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5545–5554, 2019.

[137] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In *CVPR*, 2017.

[138] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully Convolutional Geometric Features. In *ICCV*, 2019.

[139] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *arXiv preprint arXiv:1702.08734*, 2017.

[140] Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 2020.

[141] Heng Yang and Luca Carlone. A Polynomial-time Solution for Robust Registration with Extreme Outlier Rates. In *Robotics: Science and Systems XV, University of Freiburg, Freiburg im Breisgau, Germany, June 22-26, 2019*, 2019.

[142] Milad Ramezani and Kourosh Khoshelham. Vehicle positioning in GNSS-deprived urban areas by stereo visual-inertial odometry. *IEEE Transactions on Intelligent Vehicles*, 3(2):208–217, 2018.

[143] Tung Dang, Shehryar Khattak, Christos Papachristos, and Kostas Alexis. Visual-Inertial Odometry-enhanced Geometrically Stable ICP for Mapping Applications using Aerial Robots. *CoRR*, 2018.

[144] Evangelos Boukas, Antonios Gasteratos, and Gianfranco Visentin. Towards orbital based global rover localization. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2874–2881. IEEE, 2015.

[145] Evangelos Boukas, Antonios Gasteratos, and Gianfranco Visentin. Introducing a globally consistent orbital-based localization system. *Journal of Field Robotics*, 35 (2):275–298, 2018.

[146] Lazaros Nalpantidis, Georgios Ch Sirakoulis, and Antonios Gasteratos. A dense stereo correspondence algorithm for hardware implementation with enhanced disparity selection. In *Hellenic conference on Artificial Intelligence*, pages 365–370. Springer, 2008.

[147] Lazaros Nalpantidis, Georgios Ch Sirakoulis, and Antonios Gasteratos. Non-probabilistic cellular automata-enhanced stereo vision simultaneous localization and mapping. *Measurement Science and Technology*, 22(11):114027, 2011.

[148] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[149] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast Global Registration. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II*, volume 9906 of *Lecture Notes in Computer Science*, pages 766–782. Springer, 2016.

[150] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A Modern Library for 3D Data Processing. *CoRR*, abs/1801.09847, 2018.

[151] Thomas Moore and Daniel Stouch. A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In *Intelligent Autonomous Systems 13 - Proceedings of the 13th International Conference IAS-13, Padova, Italy, July 15-18, 2014*, volume 302 of *Advances in Intelligent Systems and Computing*, pages 335–348. Springer, 2014.

[152] Christos Papachristos, Frank Mascarich, and Kostas Alexis. Thermal-Inertial Localization for Autonomous Navigation of Aerial Robots through Obscurants. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 394–399, 2018. doi: 10.1109/ICUAS.2018.8453447.

[153] Konstantinos A Tsintotas, Loukas Bampis, and Antonios Gasteratos. Modest-vocabulary loop-closure detection with incremental bag of tracked words. *Robotics and Autonomous Systems*, 141:103782, 2021.

[154] Konstantinos A Tsintotas, Panagiotis Giannis, Loukas Bampis, and Antonios Gasteratos. Appearance-Based Loop Closure Detection with Scale-Restrictive Visual

Features. In *International Conference on Computer Vision Systems*, pages 75–87. Springer, 2019.

[155] Konstantinos A Tsintotas, Loukas Bampis, Antonios Gasteratos, and FIET. Tracking-DOSeqSLAM: A dynamic sequence-based visual place recognition paradigm. *IET Computer Vision*, 15(4):258–273, 2021.

[156] David Landry, François Pomerleau, and Philippe Giguere. CELLO-3D: Estimating the Covariance of ICP in the Real World. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8190–8196. IEEE, 2019.

[157] Andrea Censi. An accurate closed-form estimate of ICP's covariance. In *Proceedings 2007 IEEE international conference on robotics and automation*, pages 3167–3172. IEEE, 2007.

[158] Martin Brossard, Silvere Bonnabel, and Axel Barrau. A new approach to 3D ICP covariance estimation. *IEEE Robotics and Automation Letters*, 5(2):744–751, 2020.

[159] Fahira Afzal Maken, Fabio Ramos, and Lionel Ott. Stein ICP for Uncertainty Estimation in Point Cloud Matching. *IEEE Robotics and Automation Letters*, 7(2): 1063–1070, 2022.

[160] Contreras Juan, Cuadrado William, Munoz David, Archbold George, Delgado, Geraldine Delgado, and Diaz Vladimir. Automatic ship hull inspection using fuzzy logic. In *Proceedings - Applied Imagery Pattern Recognition Workshop*, 2012. ISBN 9781467345583. doi: 10.1109/aipr.2012.6528214.

[161] VV Kostenko, A Yu Bykanova, and A Yu Tolstonogov. Underwater robotics complex for inspection and laser cleaning of ships from biofouling. In *IOP Conference Series: Earth and Environmental Science*, volume 272. IOP Publishing, 2019.

[162] Dario Floreano and Robert J Wood. Science, technology and the future of small autonomous drones. *nature*, 521(7553):460–466, 2015.

[163] Paolo De Petris, Huan Nguyen, Tung Dang, Frank Mascarich, and Kostas Alexis. Collision-tolerant Autonomous Navigation through Manhole-sized Confined Environments. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 84–89, 2020. doi: 10.1109/SSRR50563.2020.9292583.

[164] Paolo De Petris, Huan Nguyen, Mihir Kulkarni, Frank Mascarich, and Kostas Alexis. Resilient collision-tolerant navigation in confined environments. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2286–2292. IEEE, 2021.

[165] Olivier Idir and Alessandro Renzaglia. Multi-Robot Weighted Coverage Path Planning: a Solution based on the DARP Algorithm. In *17th International Conference on Control, Automation, Robotics and Vision*, 2022.

[166] Anne Steenbeek and Francesco Nex. CNN-Based Dense Monocular Visual SLAM for Real-Time UAV Exploration in Emergency Conditions. *Drones*, 6(3):79, 2022.

[167] Zheng Fang, Shichao Yang, Sezal Jain, Geetesh Dubey, Stephan Roth, Silvio Maeta, Stephen Nuske, Yu Zhang, and Sebastian Scherer. Robust autonomous flight in constrained and visually degraded shipboard environments. *Journal of Field Robotics*, 34(1):25–52, 2017.

[168] Lorenz Meier, Petri Tanskanen, Friedrich Fraundorfer, and Marc Pollefeys. PIXHAWK: A system for autonomous flight using onboard computer vision. In *2011 IEEE International Conference on Robotics and Automation*, pages 2992–2997, 2011. doi: 10.1109/ICRA.2011.5980229.

[169] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. ROS: an open-source Robot Operating System. In *ICRA workshop on open source software*, volume 3.2, page 5. Kobe, Japan, 2009.

[170] John Wang and Edwin Olson. AprilTag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4193–4198. IEEE, oct 2016. ISBN 978-1-5090-3762-9. doi: 10.1109/IROS.2016.7759617.

[171] Bernd Pfrommer and Kostas Daniilidis. Tagslam: Robust slam with fiducial markers. *arXiv preprint arXiv:1910.00679*, 2019.

[172] Dirk Holz, Alexandru E Ichim, Federico Tombari, Radu B Rusu, and Sven Behnke. Registration with the point cloud library: A modular framework for aligning in 3-D. *IEEE Robotics & Automation Magazine*, 22(4):110–124, 2015.

[173] Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. *Advances in neural information processing systems*, 29, 2016.

[174] Eric A Wan and Rudolph Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000.

[175] Thomas Moore and Daniel Stouch. A generalized extended kalman filter implementation for the robot operating system. In *Intelligent autonomous systems 13*, pages 335–348. Springer, 2016.

[176] Simon J Julier and Jeffrey K Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.

[177] P.J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. doi: 10.1109/34.121791.

[178] Andrea Tagliabue, Jesus Tordesillas, Xiaoyi Cai, Angel Santamaria-Navarro, Jonathan P How, Luca Carlone, and Ali-akbar Agha-mohammadi. LION: Lidar-Inertial observability-aware navigator for Vision-Denied environments. In *International Symposium on Experimental Robotics*, pages 380–390. Springer, 2020.

[179] Mark Pilling and Bruce S Holden. Choosing trays and packings for distillation. *Am. Inst. Chem. Eng. CEP*, 105:44–50, 2009.

[180] Rasmus Eckholdt Andersen, Marcin Zajaczkowski, Harsh Jaiswal, Jianan Xu, Wenjie Fan, and Evangelos Boukas. Depth-based Deep Learning for Manhole Detection in UAV Navigation. In *2022 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6, 2022. doi: 10.1109/ist55454.2022.9827720.

[181] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pages 145–152. IEEE, 2001.

[182] Andreas Bircher, Mina Kamel, Kostas Alexis, Helen Oleynikova, and Roland Siegwart. Receding horizon" next-best-view" planner for 3d exploration. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1462–1468. Ieee, 2016.

[183] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous robots*, 34(3):189–206, 2013.

[184] Steven M LaValle et al. Rapidly-exploring random trees: A new tool for path planning. *Ames*, 1998.

[185] Roland Hess. *The essential Blender: guide to 3D creation with the open source suite Blender*. No Starch Press, 2007.

[186] Janosch Nikolic, Michael Burri, Joern Rehder, Stefan Leutenegger, Christoph Huerzeler, and Roland Siegwart. A UAV system for inspection of industrial facilities. In *2013 IEEE Aerospace Conference*, pages 1–8. Ieee, 2013.

[187] Ioannis Kansizoglou, Nicholas Santavas, Loukas Bampis, and Antonios Gasteratos. HASeparator: Hyperplane-Assisted Softmax. *arXiv preprint arXiv:2008.03539*, 2020.

[188] Chingiz Kenshimov, Loukas Bampis, Beibut Amirgaliyev, Marat Arslanov, and Antonios Gasteratos. Deep learning features exception for cross-season visual place recognition. *Pattern Recognition Letters*, 100:124–130, 2017.

[189] Loukas Bampis, Antonios Gasteratos, and Evangelos Boukas. CNN-based novelty detection for terrestrial and extra-terrestrial autonomous exploration. *IET Cyber-Systems and Robotics*, 3(2):116–127, 2021.

[190] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154. Ieee, 2004.

[191] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics*, pages 621–635. Springer, 2018.

[192] S. Karaman and E. Frazzoli. Incremental Sampling-based Algorithms for Optimal Motion Planning. *CoRR*, abs/1005.0416, 2010.

[193] Lydia E Kavraki, Petr Svestka, J-C Latombe, and Mark H Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4):566–580, 1996.

[194] Fadri Furrer, Michael Burri, Markus Achtelik, and Roland Siegwart. RotorS–A modular Gazebo MAV simulator framework. In *Robot operating system (ROS)*, pages 595–625. Springer, 2016.

Department of Electrical and Photonics Engineering
Automation and Control (AUT)
Technical University of Denmark
Elektrovej, Building 326
DK-2800 Kgs. Lyngby
Denmark

www.electro.dtu.dk/
Tel: (+45) 45 25 38 00
E-mail: electro@electro.dtu.dk