Real-Time localization and Topological Mapping in Buried Pipe Networks

Xiangyu Li

Submitted in accordance with the requirements for the degree of
Master of Philosophy

The University of Leeds
School of Computing

May 2024

Dedication

This thesis is dedicated to my parents, for their unwavering support and encouragement throughout my academic journey. Your belief in me has been a constant source of inspiration. I also dedicate this work to my friends and mentors who have provided invaluable guidance and support.

Declaration

The candidate confirms that the submitted work is their own, except where joint authorship has been involved. The contributions of the candidate and other authors have been explicitly indicated below.

All parts of this thesis represent the original work of the candidate, based on collaborative research with Professor Netta Cohen, Professor Mehmet Dogar, Professor Anthony Cohn, and Dr. Nguyen Luan. Their contributions involved significant input in terms of conceptual framework and technical guidance.

Xiangyu Li

Date: 2024.7.16

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Professor Netta Cohen, Professor Mehmet Dogar, and Professor Anthony Cohn, for their exceptional support, guidance, and friendship throughout my doctoral studies. Their insights and expertise have been invaluable to my research.

I am also immensely grateful to my committee member, Dr. Nguyen Luan, for his thoughtful feedback and encouragement.

Special thanks to my colleagues and friends at the University of Leeds, particularly those in the School of Computing, for their camaraderie and intellectual discussions that have enriched my academic experience.

I am forever thankful to my family for their unconditional love and support. My parents, whose sacrifices and encouragement have been my driving force, and my siblings, who have always been there for me.

Finally, I would like to acknowledge the financial support from the China Scholarship Council, which made this research possible.

Abstract

Buried pipe networks present unique challenges for robotic localization and mapping due to their constrained environments and limited accessibility. Real-time localization and mapping are essential for enabling robots to autonomously navigate these networks for inspection and maintenance. This thesis introduces methods for robotic localization and topological mapping in buried pipe networks.

The first method focuses on accurate localization by emphasising junction detection and real-time mapping. Utilising the structured nature of pipe networks, the robot identifies and maps junctions, reducing computational load and reliance on continuous video processing. A localization algorithm employing convolutional filters accurately identified junctions in the network. When navigating a simulated pipe network, the robot activated its camera only at junctions, reducing cumulative localization errors significantly. By comparing newly captured images with a pre-existing database, the robot determined whether the junction was known or new, adding it to a topological map.

Building upon this, the second method enhances mapping accuracy by using panoramic images created from junction photos. These images enabled the robot to determine the number and angles of junction exits, with the angle error maintained within 10 degrees. This method generated a topological map that provided a more accurate representation of the physical layout of the pipe network, improving navigation and planning.

Developed in a simulation environment and tested in experiments with physical robots in pipes, the second method demonstrated improved mapping precision and real-time localization, offering potential applications for the mapping, inspection, and maintenance of buried infrastructure.

Abbreviations

Abbreviation	**Definition**
AI	Artificial Intelligence
CNN	Convolutional Neural Network
FFT	Fast Fourier Transform
GPS	Global Positioning System
IMU	Inertial Measurement Unit
LiDAR	Light Detection and Ranging
NCC	Normalized Cross-Correlation
MPC	Model Predictive Control
RL	Reinforcement Learning
SLAM	Simultaneous Localization and Mapping
UKRI	United Kingdom Research and Innovation
MCL	Monte Carlo Localization
ICAIR	The Integrated Civil and Infrastructure Research Centre
ROS	Robot Operating System
IRL	Integral Reinforcement Learning
BaLL	Bayesian Landmark Learning
GVG	Generalized Voronoi Graph
SMA	Shape Memory Alloy
MAV	Micro Air Vehicle
MHT	Multiple Hypothesis Tracking
ORB	Oriented FAST and Rotated BRIEF
SIFT	Scale-Invariant Feature Transform
SURF	Speeded-Up Robust Features
HOG	Histogram of Oriented Gradients
TSDF	Truncated Signed Distance Fields
PID	Proportional-Integral-Derivative
H_∞	H-infinity (Control Theory)
AoA	Angle of Arrival
AoD	Angle of Departure
TDoA	Time Difference of Arrival
RSSI	Received Signal Strength Indicator
EKF	Extended Kalman Filter
PF	Particle Filters
ESPIRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
MUSIC	Multiple Signal Classification
RANSAC	Random Sample Consensus

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The field of robotics has experienced rapid advancements in recent years, driving innovation across sectors such as manufacturing, healthcare, infrastructure, and service industries. A particularly significant development is in autonomous robots, which can operate in environments where human intervention is difficult, dangerous, or costly. These robots are increasingly utilized for infrastructure maintenance, particularly in confined and hazardous spaces like pipes, tunnels, and underground networks. With aging infrastructure, growing urbanization, and the demand for efficient, safe, and cost-effective inspection and maintenance methods, the need for such autonomous systems is becoming increasingly urgent.

One specialized area within this field is infrastructure robotics, which focuses on deploying robots in environments such as buried pipe networks. These networks present unique challenges, including confined spaces, harsh environmental conditions, and limited access. Autonomous robots designed for these tasks must possess localization and mapping capabilities to navigate accurately and perform maintenance effectively.

A critical application within infrastructure robotics is pipe robotics, which involves the inspection, maintenance, and mapping of buried pipe networks. These robots must autonomously navigate constrained environments with limited visual data and often unreliable communication signals. While several robotic solutions have been developed for general pipe inspection, achieving real-time localization in these networks remains a significant research challenge.

Localization, the process of determining a robot's position within its environment, is vital for the success of autonomous pipe robots. The confined and featureless interiors of pipes make accurate localization particularly difficult. This thesis focuses on a specific aspect of the challenge in buried pipe networks: real-time localization and topological mapping. While general pipe robotics may address inspection or damage detection, this research narrows its scope to enhance the accuracy and efficiency of localization in environments where sensory input is limited, and traditional methods struggle.

Unlike general pipe robotics, which focuses on tasks such as inspection or damage detection, this research focuses on the problem of localization. By narrowing the scope, the aim is to enhance the robot's ability to localize and map the network accurately, even with limited sensory input. This targeted approach is expected to improve the robot's operational efficiency and contribute to safer and more reliable infrastructure maintenance. The research and development of these methods are central to the ongoing work in the Pipebots project, which is funded by UK Research and Innovation (UKRI). Pipebots are small, intelligent robots designed to operate in pipe networks, navigating autonomously to inspect, maintain, and ensure the integrity of these crucial infrastructures.

1.1 Overview of Infrastructure Robotics

Infrastructure robotics is increasingly playing a pivotal role in enhancing urban infrastructure management by integrating technologies for efficient and sustainable operations. As cities expand and their infrastructures age, the need for innovative solutions to maintain critical systems like water distribution, sewage networks, and gas supply has become more pressing. Robotics, with its ability to conduct precise inspections and perform maintenance tasks autonomously, presents a promising approach to overcoming the limitations of conventional methods. This section delves into the transformative impact of infrastructure robotics, highlighting their roles, challenges, and the specific applications within pipe networks.

1.1.1 Roles and Challenges of Infrastructure Robotics

Infrastructure robotics is increasingly becoming integral to managing and maintaining urban environments. Robotics plays transformative roles in various aspects of infrastructure management, enhancing efficiency, safety, and sustainability. Robots equipped with sensors such as cameras, LiDAR, sonar, and acoustic sensors conduct thorough inspections of infrastructure systems (Lattanzi and Miller, 2017). These sensors allow for the collection of high-resolution data, enabling detailed mapping and assessment of the structural conditions of critical infrastructure. Visual inspections identify issues like cracks, blockages, and corrosion, providing vital information for maintenance planning. Moreover, robots can perform continuous and autonomous inspections without human intervention, reducing the need for manual checks that are often labor-intensive and risky (Menendez et al., 2018). Frequent inspections help detect potential issues early, preventing major failures and minimizing disruptions (Balaguer and Abderrahim, 2008).

Infrastructure robots play a crucial role in the proactive maintenance of urban systems. They are capable of performing minor repairs, such as clearing obstructions and patching small leaks, thereby extending the lifespan of infrastructure assets. This proactive approach shifts maintenance strategies from reactive to preventive, leading to cost savings and reduced environmental impact. The integration of robotics into infrastructure management enables the collection of vast amounts of data, essential for predictive analytics and decision-making. Robots equipped with AI algorithms can analyze data in real-time, providing insights into system performance and identifying areas that require attention (Vrontis et al., 2023). This data-driven approach enhances understanding and supports the development of more efficient maintenance strategies.

Additionally, the use of edge computing and IoT in conjunction with robotics allows for seamless data integration and analysis. This capability enables utility companies to optimize asset monitoring and performance evaluation, ensuring that maintenance resources are allocated effectively. The deployment of robotic technologies contributes to the resilience of urban infrastructures (Golubchikov and Thornbush, 2020). By improving the reliability and efficiency of infrastructure systems, robotics supports sustainable urban development. This aligns with smart city initiatives that aim to integrate technologies into urban management practices, enhancing service delivery and environmental protection.

However, the integration of robotics into infrastructure systems, particularly pipe networks, presents several unique challenges. Robots operating in pipe networks face harsh and hazardous conditions such as sewer lines, water mains, and underground pipes. These environments can be wet, dark, and filled with debris, which can interfere with robotic sensors and mechanical components. The presence of water, mud, and other particulates can obscure sensor readings and cause damage to components, posing significant reliability challenges. Additionally, robots must navigate confined spaces that often have varying diameters, irregular shapes, and unpredictable layouts, making maneuverability difficult. Navigation systems capable of adapting to these unpredictable conditions are essential to cope with these challenges.

Accurate localization and mapping are crucial for robotic operations in pipe networks. Many current systems rely heavily on visual sensors, which can be hindered by poor lighting conditions and obstructions. Simultaneous Localization and Mapping (SLAM) technologies are vital for handling the complexities of underground pipes, enabling robots to autonomously navigate and map these environments (Rezende et al., 2020). Effective communication between robots and their human operators is critical, especially in underground environments where conventional wireless communication methods can be severely impeded by dense and metallic surroundings (Zhao et al., 2017). Maintaining reliable communication in environments with high signal attenuation and interference is a significant challenge, requiring robust communication protocols. Systems that rely on distance and vision sensors face challenges in environments with poor visibility or obstructed paths. Future systems need to integrate additional sensor modalities, such as thermal imaging or radar, to enhance robustness and ensure reliable operation under various environmental conditions.

From a practical perspective, scaling robotic systems to cover larger pipe networks is challenging. Systems that perform well in controlled environments may face additional challenges in larger networks with numerous junctions and varying pipe diameters. The overall energy efficiency of robotic systems during long-duration missions is a concern. Optimizing battery life and energy management strategies is essential to support extended operations in extensive pipe networks (Gurguze and Turkoglu, 2018). Real-world scenarios often involve unexpected obstacles and debris that can impact the robot's navigation. Obstacle detection and avoidance algorithms are necessary to improve reliability and safety during operations (Pandey et al., 2017).

In conclusion, infrastructure robotics plays a multifaceted role in enhancing the management and sustainability of urban environments. Through inspection, maintenance, data collection, and autonomous operation, robots contribute to the development of smarter, more resilient urban infrastructures. However, overcoming the challenges associated with pipe network robotics spanning environmental, technical, and practical domains requires advancements in sensor technologies, communication systems, and autonomous navigation capabilities, as well as strategic energy management and scalability solutions. Addressing these issues is crucial for the successful deployment of robotic systems in diverse and challenging real-world environments, ultimately supporting the creation of smarter, more connected cities.

1.1.2 Overview of Pipe Networks

Pipe networks are essential to modern urban and industrial systems, facilitating the transportation of water, gas, oil, and other fluids across extensive areas. These networks form the backbone of key utilities, ensuring that resources are delivered to homes, businesses, and industries efficiently. Constructed using durable materials such as steel, concrete, and plastic, pipe networks are designed to withstand harsh environmental conditions, pressure variations, and chemical exposure.

In the UK and across Europe, pipe networks are vast and intricate. For instance, the UK's water supply network includes approximately 354,000 kilometers of pipes, while its gas network spans about 284,000 kilometers (Gabriel, 2024; Resources, 2020). These networks are supported by extensive infrastructure, including pumping stations, reservoirs, and treatment plants, to maintain consistent service delivery. According to industry data, a significant portion of these networks, particularly in older cities, was laid in the 19th and early 20th centuries, which makes them vulnerable to deterioration and failure due to aging (Repair, 2024).

The management and maintenance of these pipe networks are critical to ensuring their continued operation. In Europe, aging infrastructure is a major concern, with many countries investing billions annually in maintaining and upgrading their networks (Repair, 2024). For example, it is estimated that water utilities in Europe spend around €45 billion annually on the operation, maintenance, and upgrading of their pipe infrastructure. Despite these efforts, issues such as pipe bursts, leaks, and blockages remain common, leading to water loss and service disruptions. In the UK alone, around 3 billion liters of water are lost daily due to leaks (Repair, 2024), highlighting the scale of the challenge.

The complexity of pipe networks, particularly in urban settings, presents unique challenges for maintenance and inspection. Conventional methods of manual inspection and repair often require excavation, which is not only costly and time-consuming but also disruptive to traffic and daily life. In response to these challenges, European countries are increasingly turning to technologies such as robotics, artificial intelligence, and remote sensing to monitor and maintain pipe networks (Resources, 2020). Robotic systems equipped with cameras, sensors, and data processing algorithms can navigate through pipes autonomously, identifying defects, mapping the internal structure, and even performing minor repairs. This shift towards automation is aimed at reducing operational costs, minimizing disruptions, and extending the lifespan of the infrastructure (Pipeline and Journal, 2024).

In addition to technological advancements, many European countries are adopting more sustainable practices in the management of their pipe networks. Efforts are being made to reduce water loss, improve the energy efficiency of pumping stations, and adopt eco-friendly materials for new installations and repairs. The push towards creating “smart” pipe networks, integrated with real-time monitoring systems, is also gaining momentum. These smart networks use sensors to continuously monitor pipe conditions, flow rates, and pressure levels, allowing for early detection of potential issues and more efficient resource management (Resources, 2020; Pipeline and Journal, 2024).

In summary, pipe networks are a critical element of modern infrastructure, supporting a wide range of utilities that are essential to daily life. In the UK and across Europe, these networks are vast and aging, necessitating significant investment in their maintenance and modernization. With the help of technologies and smart infrastructure initiatives, the future of pipe networks looks to be more resilient, efficient, and sustainable.

However, this study will focus on buried pipe networks, which feature characteristics such as junctions, manholes, and small pipe diameters, making it inaccessible for humans to enter the pipes directly. Buried pipe networks are critical components of urban infrastructure, responsible for collecting and transporting wastewater from residential, commercial, and industrial areas to treatment facilities. They play a pivotal role in protecting public health and the environment by preventing the contamination of natural water bodies. As shown in Figure 1.1, the interior of a buried pipe is often characterized by limited visibility and confined space, which complicates maintenance efforts.

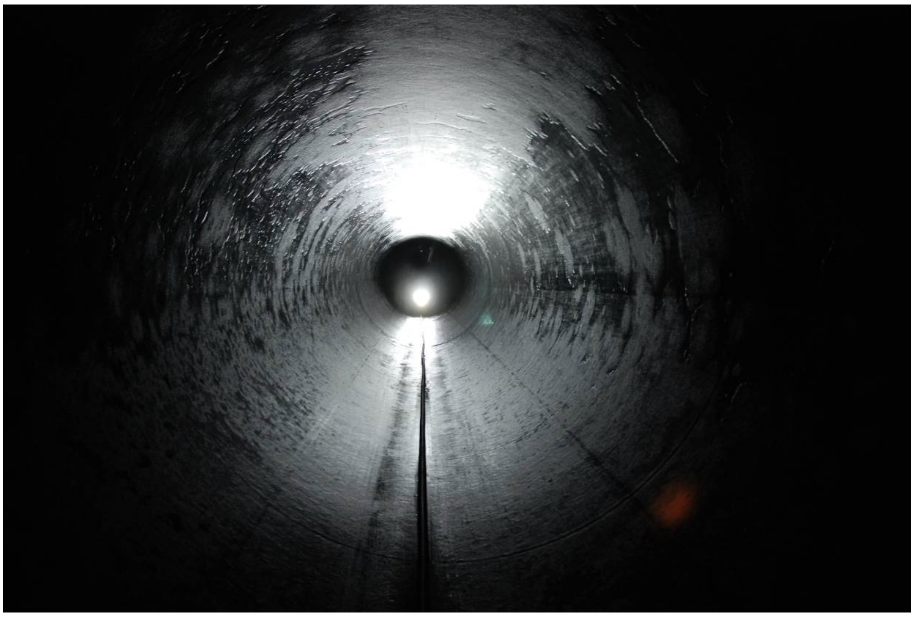


Figure 1.1: Interior view of a buried pipe, demonstrating the confined space and limited visibility that pose challenges for maintenance and inspection.

These networks are typically composed of a series of interconnected pipes and channels, designed to transport wastewater by gravity or with the aid of pumps. Key components include sewer pipes, junctions, manholes, and treatment plants. The design of buried pipe networks must consider factors such as flow capacity, pipe material, slope, and access for maintenance, despite the limited accessibility of the system.

Buried pipe networks face unique challenges, including the management of inflow and infiltration, aging infrastructure, blockages, and capacity limitations during heavy rainfall. These challenges necessitate effective design, regular maintenance, and strategic upgrades to ensure reliable and efficient operation. By focusing on buried pipe networks, this study aims to explore these challenges in detail. The internal view of a buried pipe, as seen in Figure 1.2, illustrates the structural characteristics and potential challenges, which are crucial for understanding how these systems function and the difficulties in their maintenance.

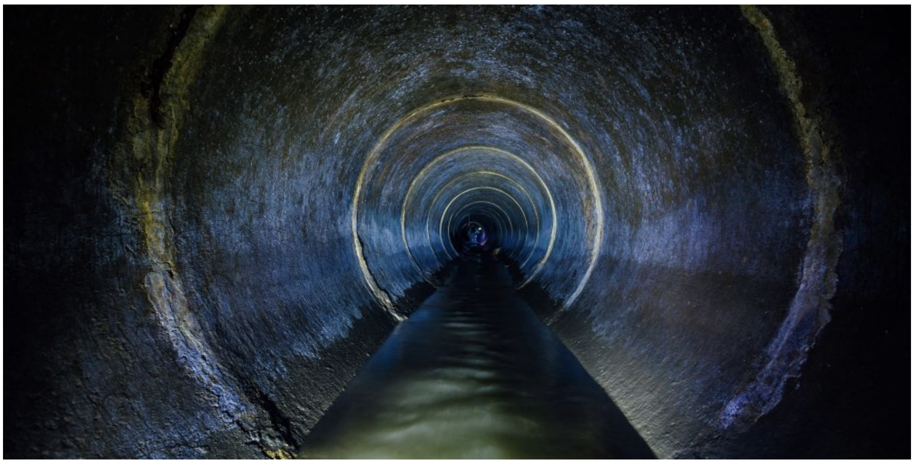


Figure 1.2: Internal structure of a buried pipe, showcasing its design and potential obstacles for effective management and maintenance.

The primary focus of this thesis is on localization and mapping within these buried networks. Accurate localization and mapping are essential for the maintenance and repair of these systems, helping to identify the exact locations of pipes, as well as any defects or blockages within them. This study will explore technologies and methodologies for achieving localization and mapping in buried pipe networks, addressing the challenges posed by the inaccessible environments within these systems. The findings aim to contribute to enhanced performance and sustainability of these networks through improved maintenance strategies.

1.1.3 Significance, and Characteristics of Buried Pipe Networks

Buried pipe networks are a critical component of modern infrastructure, providing essential services in various sectors, including water distribution, wastewater management, gas supply, and industrial processes (Moser, 1990). The design, installation, operation, and maintenance of these underground networks are vital for ensuring the continuous delivery of essential resources while minimizing disruptions to aboveground activities.

The significance of buried pipe networks lies in their hidden yet indispensable role across multiple domains (Hao et al., 2012). In water distribution systems, these networks ensure that clean drinking water is delivered to residential, commercial, and industrial areas without contamination or interruption, even under urban landscapes (Eiger et al., 1994). They also supply water for fire protection systems, contributing to public safety. In wastewater and sewage systems, buried pipe networks play a crucial role in safely transporting sewage and stormwater away from populated areas, preventing contamination of water bodies, and thus protecting public health and the environment (Ashley and Hopkinson, 2002). In the realm of gas distribution, buried pipe networks are designed to deliver natural gas and other fuels safely and efficiently to homes and industries, ensuring a reliable energy supply (Herrán-González et al., 2009). In industrial processes, underground pipes transport chemicals, process fluids, and other materials necessary for manufacturing while reducing hazards associated with surface piping systems.

The characteristics of buried pipe networks are shaped by their operational environment, which demands specific design considerations for reliability, longevity, and safety (Hao et al., 2012). These networks typically consist of pipes made from materials like steel, PVC, ductile iron, or concrete, chosen for their ability to withstand varying pressures, fluid types, and environmental conditions. Fittings, valves, and pumps are integrated into the network to manage flow direction, regulate pressure, and ensure that fluids move efficiently through the system. In buried systems, additional factors like soil conditions, corrosion, and external loads must be considered to prevent system failure over time. Proper fluid dynamics calculations are essential to avoid issues such as water hammer, which can damage pipes or cause bursts (Sharifi et al., 2018). The topology of the network is carefully planned to ensure redundancy, allowing the system to remain operational even if certain sections need maintenance or experience disruptions.

Operationally, buried pipe networks must be designed to manage pressure fluctuations that could lead to leaks or bursts, which are particularly challenging to detect and repair due to the network's hidden nature. As such, modern buried networks often incorporate technologies like sensors and automated systems for real-time monitoring of pressure, flow rates, and potential failures (Narayanan and Sankaranarayanan, 2019). These technologies enable proactive maintenance and quick response to emerging issues, reducing the risk of larger disruptions and environmental damage.

Despite their importance, buried pipe networks face numerous challenges. Aging infrastructure is a significant concern, with many systems, particularly in older cities, having been installed decades ago (Doyle et al., 2008). These aging systems are prone to leakage, which not only results in resource wastage but also poses risks to public health and environmental safety due to contamination of groundwater or other resources. Repairing and upgrading buried pipe networks can be complex and costly, often requiring excavation, which disrupts traffic and daily life (Chung et al., 2006). However, technological advancements, such as trenchless repair techniques, are helping to minimize these impacts.

Overall, the effective management of buried pipe networks is crucial to maintaining their functionality and ensuring that essential services continue to be delivered safely and efficiently. This involves regular inspection, timely repairs, emergency response planning, and long-term strategies for upgrading and expanding systems to accommodate growing demand.

1.1.4 Robots in Buried Pipe Networks

Maintaining buried pipe networks, which are essential for utilities such as water, gas, and sewage systems, is a challenge due to their underground location and the difficulty in accessing them. These networks are often hidden beneath layers of infrastructure, making inspections and repairs time-consuming and disruptive. Timely maintenance is critical to avoid blockages, leaks, and potential failures that could lead to service disruptions. Conventional methods of managing these networks typically involve reactive maintenance, where manual inspections and repairs are carried out after a failure has been detected. This often necessitates the excavation of roads and other infrastructure to access the buried pipes, causing traffic delays, economic impacts, and safety risks for workers entering confined, hazardous environments.

To address these challenges, there is a growing need for proactive, non-invasive technologies capable of predicting and mitigating potential failures in pipe networks. Advancements in robotics offer promising solutions, providing a means to perform inspections, identify faults, and carry out minor repairs in pipe systems without the need for significant human intervention. Equipped with sensors and autonomous navigation capabilities, robots can efficiently operate within pipe networks to detect issues before they escalate into larger problems, ultimately enhancing the safety and efficiency of network management (Aitken et al., 2021).

One prominent example of such technology is the Pipebots project, funded by the UK Research and Innovation (UKRI). Pipebots are small, intelligent robots designed to navigate the pipe networks of various utilities to perform tasks such as inspection, cleaning, and minor repairs. These robots are equipped with a range of sensors, including cameras, LiDAR, and ultrasonic sensors, which allow them to detect and assess defects in pipes, such as cracks, leaks, and corrosion (Team, 2019). As shown in Figure 1.3, these sensors and components enable the robot to perform autonomous navigation and operate effectively within confined environments.

The Pipebots project has successfully developed miniaturized robots that can access pipes as small as 75 mm in diameter. This capability is particularly advantageous for older or narrower pipe networks, where manual inspection would be difficult or impossible. These robots use a combination of visual and acoustic sensors to map the interior of pipes, enabling the detection of various issues such as blockages, cracks, and material degradation (Nguyen et al., 2022). The ability to generate detailed maps and provide accurate condition assessments is crucial for effective maintenance planning and decision-making.

Robots used in pipe networks can be equipped with various sensors, including cameras, sonar, LiDAR, and gas detectors, to gather a comprehensive array of data about the condition of the pipes. For instance, robots equipped with cameras

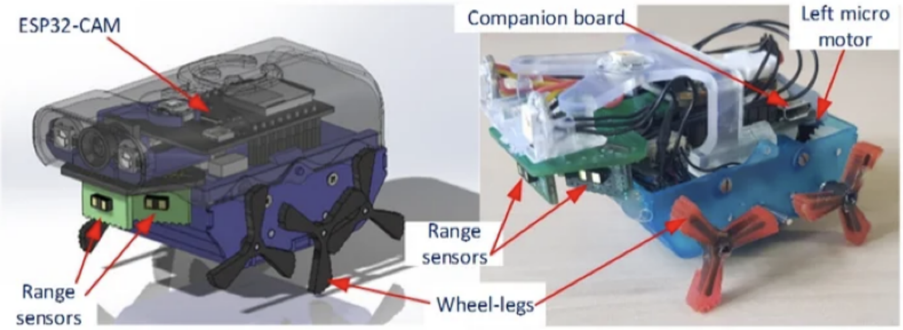


Figure 1.3: Illustration of a robot's sensors and components, showing how the robot performs autonomous navigation.

can capture detailed images of pipe interiors, allowing for the detection of cracks, blockages, and other structural issues. The Pipebots project, for example, has leveraged high-resolution cameras to conduct visual inspections and classify defects in pipe networks (Team, 2019).

In addition to visual inspections, sonar and LiDAR systems are often used to map the internal geometry of pipes and identify deformities or sediment build-up that may not be visible through cameras. Zhang et al. utilized LiDAR technology within a visual SLAM system to enhance mapping accuracy in pipe networks (Zhang et al., 2023). LiDAR provides precise distance measurements, enabling robots to create detailed 3D maps of pipe interiors, even in environments with low light or difficult conditions. Acoustic sensors can also play a role, as demonstrated by Yu et al., who utilized sound wave reflections to detect variations in pipe materials and identify blockages (Yu et al., 2023). This method is especially useful in environments where visual data is insufficient or unreliable.

Smart pipe networks are becoming key investment areas for utility companies in developed countries, supporting the transition toward more sustainable and resilient infrastructure. Advances in robotics, in conjunction with the Internet of Things (IoT) and edge computing, enable utility companies to collect more detailed information about their networks and assets. Robotic systems, operating autonomously within pipe networks, can continuously monitor conditions, shifting maintenance strategies from reactive to proactive. These autonomous robotic systems, integrated with AI and sensing technologies, can supplement conventional data sources to improve network performance analysis and maintenance planning. With access to big data from robotic inspections, utility companies can simulate network behavior, assess performance, and identify potential issues such as bottlenecks or capacity constraints (Mounce et al., 2021).

The use of robotic technologies in pipe networks also offers the potential for significant cost savings. Automated inspections can be performed more frequently and thoroughly than manual methods, leading to earlier detection of issues and reducing the risk of critical failures. This proactive maintenance approach extends the lifespan of pipe infrastructure and optimizes resource allocation for maintenance activities.

The Pipebots project is making strides in advancing micro-robotic systems for inspecting and maintaining pipe networks. These small robots are being developed to navigate the confined spaces within pipes, helping to detect and address potential issues early on, which could help reduce costs and limit disruptions to infrastructure. The project is a joint effort between academic institutions and industry partners, working together to improve the sustainability and efficiency of utility networks.

As a member of the Pipebots project group, I contribute to the development of autonomous robotic systems for localization and mapping within pipe networks. Our goal is to create intelligent, resilient systems capable of delivering failure-free operations, eliminating the need for unplanned excavations and ensuring more reliable service delivery across critical infrastructure networks.

1.1.5 Mapping and localization

This research addresses the challenges of real-time localization and mapping for robots operating in constrained environments like buried pipe networks, where conventional techniques relying on continuous visual input or GPS signals are ineffective. Instead, this system combines topological mapping with geometric localization to navigate the unique constraints of pipe networks.

Given the limitations of subterranean settings, such as low-light conditions and narrow spaces, continuous visual data collection is impractical. Therefore, the system uses selective visual sampling at critical points like pipe junctions, integrated with data from distance sensors and inertial measurement units (IMUs). The primary challenge is accurately localizing the robot based on limited visual data, while maintaining a precise map of the pipe network. This requires the development of real-time algorithms that handle image processing, localization, and mapping despite the limited computational capacity of small robotic systems.

The topological maps generated must not only reflect the network's connectivity but also represent the geometric relationships between junctions. This accuracy is essential for effective navigation and decision-making within the pipe network. The robot must autonomously identify known and unknown junctions, update its position, and determine the best navigational path based on the evolving map—all in real-time. Panoramic image stitching and normalized cross-correlation (NCC) algorithms are used to enhance the system's ability to accurately differentiate between junctions. Additionally, the robot must adapt to fluctuating environmental conditions such as obstacles, varied geometries, and changes in lighting or water levels, ensuring robust localization and mapping under adverse conditions.

This research distinguishes itself from previous efforts by moving away from continuous visual input and metric mapping, both of which are computationally intensive and less reliable in subterranean environments. Instead, it adopts selective visual sampling and topological mapping, which reduce the computational and energy demands on the robot while still maintaining high accuracy. The use of panoramic image stitching and NCC further enhances map precision, setting this approach apart from conventional SLAM methodologies.

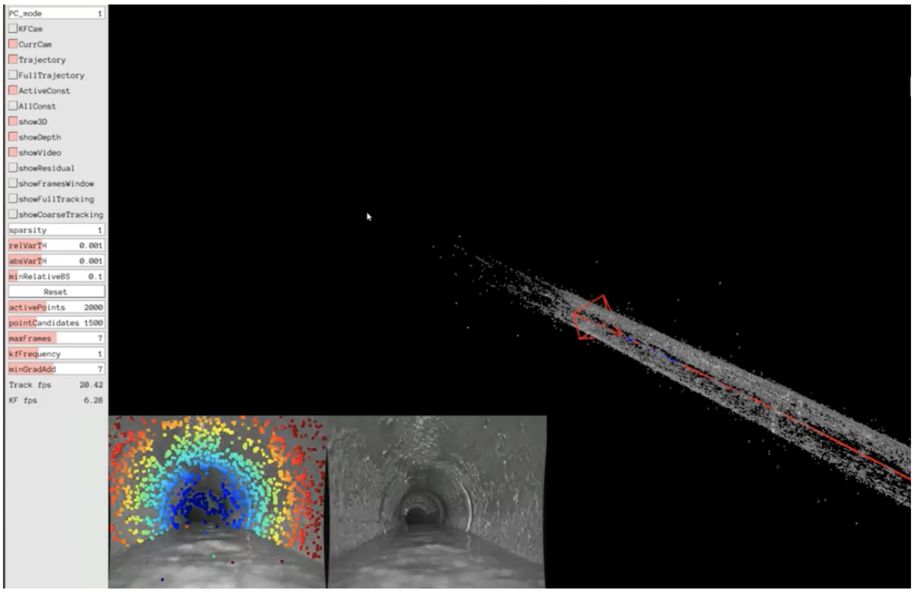
In contrast to earlier methods like visual SLAM (vSLAM) and LiDAR-based SLAM, which rely on continuous data collection and struggle in constrained environments, this system focuses on selective sampling at critical points, improving efficiency without sacrificing accuracy. SLAM technologies, especially visual and LiDAR-based, have shown great promise in many settings, but they face significant challenges when applied to confined environments like buried pipe networks. These environments are often characterized by low-light conditions, narrow passages, and featureless surfaces, which significantly degrade the performance of traditional visual SLAM systems. In such settings, conventional SLAM approaches that depend on continuous visual input are impractical due to the lack of distinguishable landmarks and the inability of cameras to capture meaningful data under poor lighting conditions. As a result, visual data often becomes noisy, causing drift and inaccuracies in the robot's estimated position over time.

Furthermore, LiDAR-based SLAM, while providing precise distance measurements, also struggles in pipe networks due to the smooth, cylindrical surfaces of pipes, which provide minimal reflective features. This limited reflection reduces the effectiveness of LiDAR in capturing accurate 3D representations of the pipe's geometry. In addition to these physical challenges, the irregular geometries of some pipe networks, with varying diameters, bends, and junctions, complicate the generation of reliable and consistent maps. The inability of traditional SLAM methods to handle such variability can lead to the failure of localization algorithms, especially in areas where the pipe network branches or where the path becomes more convoluted.

As shown in Figure 1.4, previous systems often faced difficulties in confined environments like buried pipes, where continuous data collection becomes less effective. This is primarily due to sensor limitations, such as the limited range of LiDAR and the difficulty of processing large amounts of visual data in real-time under adverse conditions. These methods also struggle with high levels of noise and interference from environmental factors, such as water accumulation, debris, or the metallic nature of pipes that can interfere with sensor signals, leading to inconsistent data and poor map quality. Furthermore, traditional systems often lack the capability to identify junctions and other critical features that are essential for accurate mapping and navigation in pipe networks. Junction recognition is a crucial component for effective localization, as it provides key reference points for the robot to update its position and continue navigating through the network.

This research addresses these challenges by incorporating NCC (Normalized Cross-Correlation) algorithms specifically designed for junction identification and differentiation. By focusing on critical points such as junctions, where the network's topology changes, the robot can generate a topological map that accurately reflects the layout of the pipe network. This method significantly reduces the reliance on continuous data collection, which is computationally expensive and less reliable in confined environments. Additionally, the selective sampling strategy ensures that the robot can operate efficiently without overburdening its limited computational and energy resources, all while maintaining high levels of mapping accuracy. By avoiding the pitfalls of continuous data collection, this system provides a more robust and resource-efficient solution for localization and mapping in underground pipe networks.

In summary, this research advances previous approaches by developing a computationally efficient system tailored to the challenges of underground pipe



```
PC_mode              1
□ KFCam
☑ CurrCam
☑ Trajectory
□ FullTrajectory
☑ ActiveConst
□ AllConst
☑ show3D
☑ showDepth
☑ showVideo
□ showResidual
□ showFramesWindow
□ showFullTracking
□ showCoarseTracking
sparsity             1
relVarTH         0.001
absVarTH         0.001
minRelativeBS      0.1
        Reset
activePoints      2000
pointCandidates   1500
maxFrames            7
kfFrequency          1
minGradAdd           7
Track fps        20.42
KF fps            6.28
```

Figure 1.4: Example of a SLAM map generated using continuous data collection, illustrating the challenges of operating in confined environments like buried pipes. The image shows a point cloud representation of a pipe network generated through visual and LiDAR-based SLAM techniques.

networks. It leverages selective visual sampling, panoramic image stitching, and topological mapping to improve localization and mapping accuracy without overburdening the robot's limited resources.

1.2 Research Motivation

The motivation for this research stems from the increasing need for improved robotic systems capable of navigating and mapping buried pipe networks autonomously and efficiently. The conventional methods of pipe inspection and maintenance, which often involve sending human workers into hazardous environments, are not only labor-intensive but also pose significant safety risks. These environments, characterized by confined spaces and challenging conditions such as poor visibility, obstructions, and water accumulation, make the application of conventional technologies difficult and inefficient.

Current robotic systems face significant limitations, particularly when it comes to real-time localization and accurate mapping in these constrained environments. Many of these systems rely heavily on visual sensors that are susceptible to degradation in low-light conditions or environments cluttered with debris and other obstructions. Additionally, existing systems often struggle with high levels of localization error, which accumulates over time, leading to inaccurate maps that limit their utility in maintenance operations.

This research seeks to address these challenges by developing methodologies for robotic localization and topological mapping that overcome the deficiencies of existing systems. By improving the efficiency and accuracy of robotic navigation in buried pipe networks, this work aims to reduce the dependency on human intervention, thereby improving safety and operational efficiency in the inspection and maintenance of urban infrastructure.

Furthermore, this research contributes to the broader field of robotics by providing methods that can be applied in other similarly confined environments, such as underground mines, tunnels, and industrial pipes. The work presented here not only holds the potential to improve the management of buried infrastructure but also aligns with the increasing demand for smarter, more resilient urban systems in the context of global urbanization and aging infrastructure.

By enhancing the capabilities of robots to autonomously inspect and map buried pipe networks, this research promises to deliver long-term benefits for infrastructure management, reducing operational costs, minimizing disruptions, and improving the sustainability of urban services.

1.3 Assumptions, Objectives and Contributions

Assumptions

This research is based on several key assumptions essential for the design and functionality of the autonomous robotic system intended for operation in buried pipe networks:

1. The robot will operate within confined environments typical of buried pipe networks. These networks are characterized by limited space and varied geometries.

2. The pipe networks are assumed to be tree-like structures, characterized as connected graphs without cycles. This tree structure simplifies the mapping process by assuming there is a single path between any two nodes without circular paths (cycles), enhancing the efficiency of the navigation and mapping algorithms. Although these networks might inherently possess directionality in practical usage (such as sewer lines where flow is unidirectional), for the purposes of this research, the network is treated as an undirected acyclic graph where the robot can navigate in any direction irrespective of the actual flow direction.

3. The robotic system is equipped with sensors capable of estimating distances.

4. While the system is assumed to have robust real-time processing capabilities, with hardware and software optimized to manage the computational demands of image processing, mapping, and autonomous decision-making without delays, it is also designed to minimize energy and computational demands.

5. The robot is designed to either navigate independently within the pipe network, identifying junctions and making decisions autonomously without human intervention, or to be tele-operated by a human.

Objectives

The primary objectives of this research are to develop methodologies that improve real-time robotic localization and topological mapping within confined environments like buried pipe networks. The focus is on addressing the limitations of traditional SLAM methods and enhancing the efficiency of localization despite the challenging conditions such as low light and limited visual features. The key objectives include:

1. To develop a localization system that integrates distance sensing and selective visual sampling to accurately identify junctions and map the topology of the pipe network. Specifically, "localization" refers to the process by which the robot determines its position within the network based on the data collected from the sensors, enabling it to update its map and make navigational decisions.

2. To design the system to minimize energy and computational demands, ensuring that the robot can operate efficiently over extended periods. Although the focus on energy minimization was not fully supported by subsequent tests in this study, this objective remains crucial for future improvements. However, for the current research, the emphasis was primarily on ensuring the system's robustness and efficiency under constrained computational resources.

3. To validate the developed methodologies through extensive simulations and to test key aspects of the system in simulated and controlled real-world scenarios, focusing on representative situations that occur in buried pipe networks. While extensive simulation and real-world evaluations were not fully carried out, this research tests critical aspects such as junction detection and topological mapping accuracy under conditions that closely resemble those encountered in practical applications.

The contributions of this research provide practical advancements in real-time localization and topological mapping of buried infrastructure, specifically addressing challenges in junction-based localization and network representation within pipe environments.

1. Developed a localization approach that integrates distance sensing with limited visual sampling at junctions. This technique leverages the NCC algorithm for image matching to distinguish junction geometries, enhancing the robustness of topological mapping without relying on continuous video processing. This targeted approach reduces computational load and energy consumption, making it feasible for small-scale robotic platforms in constrained environments.

2. Designed a topological mapping framework tailored for pipe networks, which not only encodes connectivity but also incorporates angular relationships between nodes. The use of panoramic image stitching at junctions enables the creation of metrically enhanced maps, improving spatial representation without requiring a full metric reconstruction of the entire network.

3. Validated the proposed methodologies through controlled simulations and real-world experiments in representative pipe environments. The results confirm the system's ability to accurately identify junctions and construct a functional topological map, demonstrating its applicability within realistic operational constraints.

This research provides a focused contribution by addressing specific challenges in localization and mapping for buried pipe networks, rather than attempting to solve the broader problem of robotic navigation in complex underground infrastructures. The developed methods lay a practical foundation for future improvements in robotic autonomy for infrastructure inspection and maintenance.

1.4 Structure of the Dissertation

This dissertation is structured to provide an exploration of the research undertaken, from foundational concepts and literature review to detailed methodologies, experimental results, and conclusions. Chapter 2 reviews existing literature and background information in buried pipe networks, their maintenance challenges, and the role of robotics, highlighting current localization and mapping technologies and their limitations. Chapter 3 details the development of the autonomous control system, the use of normalized cross-correlation (NCC) for image stitching, and the implementation of active vision strategies, presenting experimental results from both simulations and real-world tests. Chapter 4 focuses on the panoramic image stitching method for topological mapping, outlining the process, determination of spatial relationships, and construction of topological maps, with experimental results validating the approach. Chapter 5 summarises the key findings and contributions, discusses limitations, and suggests directions for future research, discussing the impact of the developed methodologies on robotic navigation and mapping in buried pipe networks.

Chapter 2

Background & Literature Review

This chapter provides a critical overview of the research context, focusing on the key challenges and methodologies related to robotic localization and mapping within buried pipe networks. It also synthesizes relevant literature to highlight existing approaches and identify gaps in the field. The chapter is structured as follows: Section 2.1 introduces Simultaneous Localization and Mapping (SLAM), a fundamental technique for autonomous robots navigating and mapping confined environments such as buried pipe networks. Section 2.2 discusses the specific robotic challenges encountered in pipe network environments. Section 2.3 reviews current methodologies and technologies used for pipe robot localization and mapping. Section 2.4 explores various topological mapping techniques tailored to these unique environments. Finally, Section 2.5 covers essential aspects of autonomous control, which is crucial for efficient operation within constrained settings.

2.1 Introduction to Simultaneous Localization and Mapping (SLAM)

Simultaneous Localization and Mapping (SLAM) is a fundamental technique in robotics that allows a robot to simultaneously construct a map of an unknown environment and localize itself within that map (Durrant-Whyte and Bailey, 2006). The problem of SLAM arises when a robot must navigate an environment where external positioning systems such as GPS are unavailable, and it must rely solely on internal sensors for both navigation and mapping (Thrun, 2002). In these situations, SLAM enables a robot to explore and map its surroundings autonomously while continuously updating its position in real-time.

The core challenge of SLAM is that it involves solving two interdependent problems simultaneously:

1. **Localization** — Determining the robot's position and orientation relative to its surroundings (Dellaert et al., 1999).

2. **Mapping** — Building an accurate map of the environment based on sensor data (Thrun and Montemerlo, 2006).

Both tasks are tightly coupled; for instance, an accurate map helps improve localization, and accurate localization helps refine the map. However, the difficulty in SLAM lies in the fact that both the robot's location and the environment map are initially unknown, and errors in either can propagate, degrading both (Cadena et al., 2016). These challenges are exacerbated by sensor noise, environmental uncertainty, and dynamic changes in the surroundings (Tan et al., 2013).

In typical robotic applications, SLAM is employed using a combination of sensors such as cameras, LiDAR (Light Detection and Ranging), sonar, and Inertial Measurement Units (IMUs). These sensors help the robot perceive its environment and track its movement. The sensor data is then used by SLAM algorithms to update the robot's position and the map incrementally as the robot moves through the environment (Engel et al., 2014).

SLAM plays a critical role in environments where traditional external localization systems like GPS cannot be used. For example, in indoor environments, underground pipe networks, and other confined spaces, SLAM enables robots to navigate and map their surroundings without relying on external signals (Mur-Artal et al., 2015). This is particularly important in autonomous systems used for infrastructure inspection, exploration, and monitoring, where GPS is not available.

The key challenge in SLAM is managing the uncertainty in sensor data and handling the cumulative errors that can occur over time. As robots explore a space, errors in positioning can accumulate, leading to "drift" in the map (Olson, 2009). Several SLAM techniques have been developed to address these challenges, including probabilistic methods like Kalman Filters and Particle Filters (Kalman, 1960), which attempt to estimate the robot's position and map while accounting for uncertainty and minimizing error propagation (Fox et al., 2001). In addition, optimization-based SLAM techniques, such as GraphSLAM, are used to refine both the robot's path and the map globally by minimizing the overall error (Thrun and Montemerlo, 2006).

SLAM has been extensively studied and applied in various domains, including autonomous vehicles, robotic exploration, and mobile robotics (Cadena et al., 2016). In confined or dynamic environments such as underground pipe networks, SLAM is especially valuable, as it allows robots to perform tasks such as inspection, maintenance, and mapping autonomously and efficiently (Aitken et al., 2021).

Recent advancements in SLAM have also incorporated sensor fusion, deep learning techniques, and hybrid approaches that combine different sensor modalities to improve performance in challenging environments (Deng et al., 2019). These advancements allow SLAM to be applied more effectively in complex, cluttered, or featureless environments where traditional methods may struggle (Song et al., 2022).

In the context of underground pipe networks, SLAM plays a pivotal role by enabling robots to map the network, navigate through narrow or obstructed pathways, and detect issues such as leaks or blockages (Aitken et al., 2021). This application is critical for infrastructure maintenance, as it reduces the need for manual inspections and enhances the efficiency and safety of operations (Wang et al., 2023).

2.2 Robotic Challenges in Pipe Networks

Localization and mapping underground pipe networks pose significant challenges for robotic systems, primarily due to the harsh environmental conditions, unpredictable pipe geometries, and stringent operational demands. This section addresses the key difficulties encountered by both human-operated and autonomous robots in these environments, emphasizing issues that have shaped current approaches in the field.

2.2.1 Challenges for Human Operated Robots

Operating robots in buried pipe networks involves significant challenges, particularly when human operators are involved. Harsh conditions, such as poor lighting, dampness, and varying pipe materials, hinder both movement and sensor accuracy. Ambiguous or noisy sensor data makes it more difficult for operators to make informed decisions, and control becomes even more complex in slippery conditions where robots may drift due to fluid flow. Additionally, varying pipe profiles, such as encrustation or damage, further complicate navigation.

Roh et al. identify the difficulty of maintaining accurate location and orientation in pipes that feature bends, vertical sections, and junctions. With no access to external localization tools like GPS, operators rely on internal sensors (e.g., CCD cameras, odometry), which are prone to drift over time, complicating human control and mapping efforts. The lack of distinguishing features in pipe interiors makes it challenging to create accurate maps and navigate efficiently, emphasizing the need for simultaneous localization and mapping (SLAM) algorithms to support operators (Roh and Choi, 2005).

Environmental factors within the pipes exacerbate operational difficulties. Tight, irregular spaces and slick or debris-covered surfaces impair robot movement and obstruct sensors, leading to misinterpretations of data. Humidity, hazardous materials, and communication difficulties due to signal interference from pipe materials further complicate the situation. Solutions such as tethered communication systems and adaptive control strategies are often required, though these increase the complexity of human control (Bandala et al., 2019).

Additionally, environmental conditions like high temperatures, inert gases, and pressure variations affect the robot's mechanical and thermal performance. Boudjabi et al. discuss the difficulties of controlling robots in such environments, especially when using actuators like shape memory alloys, which exhibit nonlinearities and hysteresis (Boudjabi et al., 2003). The integration of multiple sensors and actuators to improve performance further complicates manual control, often leading to suboptimal results.

The need for reliable communication is crucial in these environments. As Doychinov highlights, wireless communication often fails due to interference from the physical composition of the pipes. Solutions like mesh networks and specialized wireless systems are necessary to maintain connectivity, ensuring real-time communication and command delivery (Doychinov et al., 2021).

Finally, sensor reliability is critical in environments that present unpredictable conditions, such as water, mud, and sediment, which can impair sensor readings.

While advancements like microphone arrays can help improve real-time decision-making, they add cognitive burdens on operators, who must interpret and adjust the robot's actions accordingly. This highlights the need for AI technologies to assist in data processing and automate decision-making, reducing the cognitive load on operators (Yu et al., 2023).

In conclusion, the challenges of human-operated robots in buried pipe networks, including harsh environmental conditions, sensor limitations, and unreliable communication, underscore the need for advanced technologies such as AI, SLAM algorithms, and robust communication systems to improve robot operation and reduce the reliance on human intervention.

2.2.2 Additional Challenges for Autonomous Robots

Deploying autonomous robots in buried pipe networks presents significant challenges due to unpredictable environments, necessitating solutions for effective navigation, maintenance, and data collection. Autonomous robots must be equipped with real-time control systems and decision-making algorithms that allow them to adapt to harsh and dynamic conditions, including darkness, humidity, and debris. The development of control algorithms is critical for enabling robots to process sensory data efficiently to execute their tasks autonomously.

A primary challenge for these robots is navigating the complex layouts of buried pipe networks. These pipes often have narrow, irregular shapes and varying diameters, making maneuvering difficult. SLAM technologies play a vital role in addressing these navigational issues, enabling robots to autonomously create maps and localize themselves within the environment. SLAM also allows for the identification of structural issues such as leaks or blockages, facilitating continuous monitoring of the infrastructure and reducing the need for manual inspections (Aitken et al., 2021).

Robots like ARSI and KANTARO are designed to function in extreme conditions, such as low visibility and high humidity. ARSI utilizes a Micro Air Vehicle (MAV) with RGB-D cameras to autonomously navigate while capturing high-definition imagery for offline analysis (Chataigner et al., 2019). KANTARO employs a combination of laser and visual sensors to inspect and navigate pipes in real-time, using infrared sensors for obstacle avoidance (Nassiraei et al., 2007). These systems demonstrate how sensor fusion and advanced algorithms enhance the capability of autonomous robots in challenging environments.

One significant limitation of autonomous robots is their battery life, which is often constrained by the energy demands of sensors and computational systems. For example, KANTARO faces energy constraints due to the power consumption of its sensors and propulsion systems, which limits its operational time. To address this challenge, energy-efficient design strategies are crucial. Selective activation of vision sensors at critical points can conserve energy and extend battery life.

Another major obstacle is multi-robot coordination, particularly in systems like Pipebots, which rely on swarm intelligence for large-scale inspections. While Pipebots share real-time data to improve fault detection and ensure efficient coverage, communication issues persist due to signal attenuation and interference within the underground environment. The use of virtual pheromones for navigation helps mitigate some of these issues, although ensuring uninterrupted communication remains a challenge (Parrott et al., 2020).

The reliance on sensors and real-time data processing introduces additional limitations. For instance, robots such as the In-Pipe Inspection Robot rely on sensor fusion techniques using contact sensors and inertial measurement units (IMUs) to maintain precise localization. However, these sensors have limitations, particularly when issues arise outside of their immediate vicinity. The computational demands of processing real-time data also contribute to delays in decision-making, which can reduce overall efficiency (Jang et al., 2022).

In response to these challenges, our robot systems are designed with modular components for easy replacement and upgrading, reducing downtime and improving overall performance. By generating topological maps instead of detailed models, our robots reduce the computational load, simplifying navigation and improving efficiency. This approach not only reduces power consumption but also increases adaptability to varying conditions, making the robots more effective for long-term use in underground infrastructures.

In summary, the deployment of autonomous robots in buried pipe networks faces several challenges, including navigation complexity, power limitations, and multi-robot coordination. Through energy-efficient designs, modular components, and advanced algorithms, these robots are increasingly capable of performing inspection and maintenance tasks in challenging environments.

2.3 Pipe Robot Localization and Mapping

This section explores the core methods of robotic localization and mapping. These methods are crucial for the inspection, maintenance, and operation of robotic systems.

2.3.1 Localization Methods

Localization methods for robotic systems are crucial in environments where GPS is unavailable, such as buried pipe networks, where robots must navigate and locate themselves precisely. Several approaches have been developed, each using various sensor technologies and algorithms to estimate a robot's position. Despite their differences, these methods share underlying principles that allow them to be grouped into categories based on their techniques. Below is a detailed exploration of these localization methods, organized by their commonalities and sequential progression in complexity and capability, with appropriate references to key literature.

Probabilistic filtering techniques, including Kalman Filters and Particle Filters, are widely employed in localization because of their effectiveness in non-static environments and their ability to cope with sensor noise. One prominent example is the work by Roumeliotis et al., who integrate Bayesian estimation with Kalman filtering to improve both accuracy and reliability in mobile robot localization. This method combines continuous displacement estimates from proprioceptive sensors, such as wheel encoders and gyroscopes, with discrete landmark observations gathered by exteroceptive sensors, such as laser range finders. By applying Multiple Hypothesis Tracking (MHT), the system maintains several possible hypotheses about the robot's location, refining these hypotheses as the robot moves through its environment and receives more sensor data. This approach allows the robot to overcome initial uncertainties and sensor noise, making it effective in non-static and noisy environments, such as offices or underground pipes, where conventional methods might struggle (Roumeliotis and Bekey, 2000).

Similarly, Fox et al. propose Monte Carlo localization (MCL), a particle filter-based method that builds upon probabilistic filtering principles. MCL estimates the robot's pose by representing possible positions with a set of particles, which are sampled and assigned importance weights based on the robot's motion and sensor data. The method excels in various localization scenarios, including position tracking, global localization, and the "kidnapped robot" problem, which involves a robot suddenly being relocated without any awareness of its new position. MCL's efficiency comes from its ability to represent multiple hypotheses and focus computational resources on the most likely ones, making it superior to grid-based methods in both accuracy and computational efficiency (Fox et al., 2001).

Simultaneous Localization and Mapping (SLAM) techniques offer a more integrated approach, enabling robots to construct a map of their environment while simultaneously localizing themselves within it. Su et al. introduce a hybrid SLAM method that integrates both lidar and visual features for robust global localization. The system begins by generating a two-dimensional occupancy grid using lidar scans, which is enriched with visual keyframes captured by an RGB-D camera. These keyframes are encoded with a Gist global descriptor for fast retrieval of pose proposals by comparing the current view with previously stored scenes. For pose refinement, local ORB (Oriented FAST and Rotated BRIEF) descriptors are used to match features between the current view and candidate keyframes, eliminating mismatches through RANSAC (Random Sample Consensus). This hybrid approach is particularly effective in addressing the "kidnapped robot" problem, as it incorporates a re-localization trigger mechanism based on pose consistency and keyframe clustering. The combination of precise geometric data from lidar and rich contextual information from visual sensors enhances localization accuracy and robustness, even in geometrically challenging environments (Su et al., 2017).

An alternative SLAM approach is presented by Choset et al., who focus on topological localization. This method relies on constructing a topological map of the environment, such as a Generalized Voronoi Graph (GVG), which represents the spatial relationships between obstacles and free space, rather than explicit coordinates. The robot localizes itself by navigating through a network of nodes, referred to as "meet points", which are equidistant from surrounding obstacles. The method includes hierarchical localization strategies: zero-dimensional methods for identifying unique node signatures, one-dimensional methods for analyzing sequences of edges and nodes, and two-dimensional methods for matching graph structures. This topological approach is advantageous in changing environments with sensor noise because it leverages the structure of the environment, allowing the robot to localize itself based on spatial relationships rather than relying solely on precise metric coordinates (Choset and Nagatani, 2001).

Landmark-based localization is another critical method, providing robots with the ability to identify and utilize distinctive environmental features for improved positioning. Thrun introduces the Bayesian Landmark Learning (BaLL) algorithm, which enables robots to autonomously learn and select the most effective landmarks from sensor data. This method employs artificial neural networks to extract low-dimensional features from high-dimensional sensor data, optimizing these features using Bayesian analysis. The BaLL algorithm dynamically refines the robot's beliefs about its location by integrating sensor observations and motion updates, minimizing localization errors over time. This approach contrasts with conventional methods that rely on static, human-defined landmarks, offering greater flexibility and adaptability to varying environments. BaLL has been shown to outperform conventional methods by providing more accurate and reliable localization across a range of scenarios (Thrun, 1998a).

Sensor-based localization methods expand upon these approaches by leveraging various sensory inputs for positioning. Amjad et al. present a detailed overview of Radio SLAM, which uses radio signal sensors to collect environmental information and estimate the positions of both the robot and environmental landmarks. Radio SLAM utilizes multipath propagation components such as Angle of Arrival (AoA), Angle of Departure (AoD), Time Difference of Arrival (TDoA), and Received Signal Strength Indicator (RSSI) to determine the positions of the robot and the landmarks. Statistical techniques such as Multiple Signal Classification (MUSIC) and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPIRIT) are employed to enhance localization accuracy. In addition, filters like the Extended Kalman Filter (EKF) and Particle Filters (PFs) are used to predict the robot's state, further assisting in precise localization and navigation. Radio SLAM is especially useful in environments where visual SLAM may be less effective, such as low-light conditions or privacy-sensitive areas, making it a valuable alternative in certain scenarios (Amjad et al., 2023).

In addition to the aforementioned probabilistic filtering, SLAM, and sensor-based localization techniques, recent advances in feature matching and deep learning approaches have also shown promise in robotic localization tasks, especially in complex environments like buried pipe networks. While traditional methods, such as Kalman Filters and Particle Filters, have dominated the field, modern techniques leverage advancements in machine learning to improve robustness and accuracy.

Feature Matching has become an essential component in many localization strategies, especially for vision-based systems. Methods like SIFT (Scale-Invariant Feature Transform) and SURF (Speeded-Up Robust Features) are commonly used for matching distinctive visual features between consecutive frames or keyframes in SLAM systems. Recent studies, such as those by Lowe (2004) and Bay et al. (2006), have enhanced the efficiency of these techniques, making them applicable to various robotic tasks, including localization within confined spaces. These methods work by detecting and matching local features that are robust to scale, rotation, and illumination changes, providing valuable data for pose estimation. Additionally, ORB (Oriented FAST and Rotated BRIEF) descriptors, often used in visual SLAM, have been employed for better computational efficiency and robustness in environments with limited visual textures (Rublee et al., 2011).

A comparative study by Bayraktar and Boyraz (2017) analyzed various combinations of feature detectors and descriptors, including SIFT, SURF, and ORB, in the context of mobile robot localization. The study found that the SIFT-SURF combination achieved the highest accuracy, while the ORB-BRIEF combination offered the fastest processing time.

Incorporating deep learning into localization systems has opened new possibilities for enhanced feature matching and scene understanding. Recent approaches have utilized Convolutional Neural Networks (CNNs) (LeCun et al., 1998) to extract high-level semantic features from images, improving the matching process in feature-poor environments. For instance, Recurrent Neural Networks (RNNs) (Rumelhart et al., 1986) and Long Short-Term Memory (LSTM) (Graves and Graves, 2012) networks are employed to learn spatial-temporal dependencies and handle challenging dynamic environments, such as those found in underground pipe networks. These methods have demonstrated increased resilience in complex, noisy, and cluttered environments where traditional techniques might struggle.

Hu et al. (2022) explores the integration of deep learning techniques to improve the accuracy and efficiency of pipeline recognition and localization using Ground Penetrating Radar (GPR). By employing a Faster R-CNN model optimized with Attention-guided Context Feature Pyramid Networks (ACFPN), the authors developed an automated system that achieved an impressive average precision of 0.9256. The approach also incorporates a positioning model using Tesseract OCR to determine the exact location of buried pipelines, both horizontally and in terms of depth. On-site experiments conducted with real-world embedded pipes demonstrated the system's effectiveness, with localization errors kept under 11 cm. This work highlights the potential of deep learning for improving pipeline detection and mapping, offering an automated, accurate, and efficient alternative to traditional methods used in subsurface utility management.

Karimanzira (2023) introduces a novel approach for detecting and localizing leaks in water distribution networks (WDNs) using a deep learning-based autoencoder model. The proposed method employs a hybrid architecture, combining a 3D Convolutional Neural Network (3DCNN) encoder with a Convolutional Long Short-Term Memory (ConvLSTM) decoder, and incorporates spatial and temporal attention mechanisms to enhance the accuracy of leak localization. The model is trained using water pressure and flow rate data, and its performance is evaluated with the Water Network Tool for Resilience (WNTR) simulator. The results show that the model detects leaks with 96% accuracy and a false positive rate of just 4%, outperforming traditional methods like random forests. This method provides an efficient and reliable solution for real-time leak detection and localization, offering significant improvements in the management of water distribution systems.

End-to-end deep learning models, like DeepVO (Deep Visual Odometry) (Wang et al., 2017), also offer promising solutions for visual localization by directly learning the relationship between consecutive frames. These models bypass the need for handcrafted features like SIFT or ORB, relying on a CNN-based architecture that learns feature correspondences and relative poses from raw sensor data. Such methods, though computationally intensive, have been shown to outperform traditional methods in certain settings, particularly when large amounts of data are available for training.

The introduction of Deep Reinforcement Learning (DRL) into localization tasks has been explored in recent research, where a robot learns to localize itself and map its surroundings by receiving feedback from its environment (Chen et al., 2020). These approaches involve training a model to make localization decisions based on sensory inputs, which has the potential to be more adaptable and robust in highly dynamic and uncertain environments.

2.3.2 Mapping Methods

Effective mapping methods are crucial for the successful operation of robots in buried pipe networks. These methods allow for the creation of accurate and detailed maps of the underground environment, which are essential for navigation, inspection, and maintenance tasks. A variety of techniques have been developed to tackle the unique challenges of mapping in these constrained environments, including laser scanning, acoustic sensing, and visual SLAM.

One of the mapping techniques is the real-time algorithm for mobile robot mapping developed by Thrun et al. This method addresses the simultaneous localization and mapping (SLAM) problem using a combination of fast scan-matching techniques and a sample-based probabilistic framework. The approach relies on 2D laser range finders to incrementally construct maps while calculating the posterior distribution of robot poses over time. The algorithm uses maximum likelihood estimation for determining robot positions and integrates posterior estimation to maintain a probabilistic representation of possible locations. This robust representation allows the algorithm to handle environments with loops and multi-robot mapping, where robots collaborate to build a unified map. Moreover, Thrun et al. extend their method to 3D mapping, employing a dual-laser system and a multi-resolution algorithm, making it adaptable for real-time navigation and exploration in various environments, even in the absence of reliable odometric data (Thrun et al., 2000).

Another approach is the EchoSLAM algorithm by Krekovic et al., which uses acoustic echoes for SLAM. This method is particularly well-suited for environments where conventional sensors may be less effective. EchoSLAM operates with a simple setup of a single omnidirectional sound source and receiver, using echoes as virtual landmarks to estimate room geometry and robot trajectory. Through a Bayesian framework, the algorithm calculates times of arrival (TOAs) of the echoes, allowing for iterative refinement of both the robot's position and the boundaries of the environment. This acoustic-based approach demonstrates the potential of non-conventional sensing modalities in SLAM applications (Kreković et al., 2016).

In the realm of visual SLAM, Naseer et al. developed a robust method utilizing visual data from cameras. This approach involves capturing images and extracting distinctive features using dense HOG (Histogram of Oriented Gradients) descriptors, which are then associated across multiple images to build a data association graph. The process models spatial relationships and is treated as a minimum cost flow problem, optimizing the path through the graph to handle challenges such as occlusions or route deviations. This visual SLAM method does not rely on GPS or odometry, making it versatile for different environments and adaptable to changing conditions (Naseer et al., 2014).

The integration of multiple sensing modalities, such as LiDAR, acoustic sensors, and visual cameras, further enhances the robustness and accuracy of mapping in buried pipe networks. Combining data from these diverse sources enables the creation of comprehensive and reliable maps by leveraging the strengths of each sensor type while compensating for their individual limitations. Projects like Pipebots exemplify this multi-modal approach, using a combination of these methods to improve mapping accuracy and reliability in underground environments, such as buried pipe networks (Team, 2019).

The mapping methods for robotic systems, particularly in confined environments like buried pipe networks, have also evolved with the incorporation of deep learning and advanced feature matching techniques. Traditional mapping methods, such as laser-based SLAM and acoustic-based SLAM, are complemented by methods that integrate visual feature matching and machine learning for improved robustness.

In the domain of visual SLAM, several deep learning-based techniques have been proposed to enhance map creation. For example, semantic segmentation models, such as DeepLab (Chen et al., 2017), have been employed in combination with traditional SLAM methods to generate semantic maps that contain both spatial and contextual information about the environment. These semantic maps can significantly improve the robot's ability to understand and navigate its environment, particularly in complex, underground spaces where traditional mapping techniques might lack the precision needed.

A promising approach that has gained attention is DeepFusion (Li et al., 2022), a method that integrates LiDAR and visual data through deep learning to build accurate and detailed 3D maps. This technique uses a deep neural network to fuse the complementary information from both sensor types, ensuring that each sensor's limitations are mitigated by the other. For instance, visual data can provide rich texture and depth information in environments where LiDAR may struggle, while LiDAR can help mitigate challenges faced by vision-based systems, such as poor lighting or occlusions. This fusion approach has been successfully applied in various autonomous navigation systems.

Additionally, learning-based feature matching techniques have been applied to map generation. For example, the use of deep feature matching networks enables robots to more reliably match features across various viewpoints, especially when the environment lacks distinct visual landmarks. These methods can be seen as extensions of traditional feature matching techniques like SIFT, but they use deep learning models to improve accuracy and robustness in feature extraction and matching (Zhou et al., 2017).

To enhance the accuracy of maps in constrained environments, techniques such as point cloud registration (Pomerleau et al., 2015) and semantic 3D reconstruction (Cherabier et al., 2018) have been explored. These methods often rely on deep learning models to classify and refine the quality of the 3D data generated by sensors, ensuring that the resulting maps are not only spatially accurate but also semantically meaningful.

2.4 Topological Mapping Methods

Topological mapping methods prioritize the representation of an environment based on its connectivity and the relationships between different locations, rather than focusing on precise geometric details (Thrun, 1998b). These methods are especially relevant in the environments such as underground pipe networks, where the primary concern is the connectivity of the network rather than exact spatial measurements.

A common approach to topological mapping is the use of graph-based methods, where the environment is represented as a graph (Brunskill et al., 2007). In this representation, nodes correspond to significant locations, such as junctions or intersections in a pipe network, while edges denote the paths or connections between these points. This abstraction simplifies the representation of the environments, facilitating tasks like navigation and path planning.

One noteworthy framework in topological mapping is the Topo map system, developed by Blöchliger et al., which enhances robot navigation by transforming sparse visual SLAM maps into efficient topological maps. These are particularly suited for large-scale environments (Blochliger et al., 2018). The method begins with the creation of a sparse feature-based map using a monocular camera to capture essential 3D landmarks. From this sparse point cloud, occupancy information is extracted using Truncated Signed Distance Fields (TSDF), which identify free and occupied spaces. The environment is then segmented into convex free-space clusters, within which the robot can move freely without encountering obstacles. Clusters are merged iteratively, simplifying the topological map by combining neighboring clusters with minimal obstacles. These clusters form the vertices of a topological graph, while portals represent transitions between areas, facilitating efficient path planning through a navigation graph using algorithms like A*. This approach significantly reduces computational load and storage requirements, making it ideal for mobile platforms with limited resources.

In a different approach, Zhang et al. proposed a topological mapping method specifically designed for multi-robot exploration in communication-restricted environments (Zhang et al., 2022). This method constructs maps using vertices and edges, where vertices represent specific areas described by visual descriptors, and edges denote the paths between them. Robots equipped with panoramic cameras generate new vertices when current observations do not match existing descriptors, reducing data transmission between robots by up to 90%. Newly created vertices and edges are shared among robots, allowing for effective collaboration while minimizing communication loads. Maps are merged by matching vertex descriptors, and robots dynamically update the topological map as they explore new areas, maximizing coverage and minimizing redundancy.

In pipe networks, Zhang et al. further enhanced SLAM by exploiting the cylindrical regularity of pipes. They combined topological maps with geometric information derived from the cylindrical structure of the pipes, enabling improved robot localization and navigation (Zhang et al., 2023). The integration of geometric data into topological maps allows for improved localization and navigation, especially in environments with cylindrical geometries such as sewer systems. However, this hybrid method inherently increases the computational complexity, as the process of fusing topological and metric data is more demanding than traditional methods. The added data processing overhead may limit real-time application, particularly in scenarios requiring high-frequency decision-making and dynamic environmental adaptation. For instance, while methods like (Choset and Nagatani, 2001) focus on simplifying computational load through pure topological maps, our approach offers improved accuracy but at the cost of increased computational demand. To mitigate this, optimization techniques, such as selective sensor data fusion and real-time processing strategies, could be explored to balance performance and computational efficiency.

Yu et al. introduced a method that uses microphone arrays to facilitate simultaneous condition detection, localization, and classification within sewer systems (Yu et al., 2023). Acoustic data from the arrays is used to build a topological map of the environment, identifying critical features such as blockages and lateral connections. This method proves effective in scenarios where visual data may be limited, offering a robust alternative for navigation and inspection in dark, murky, or obstructed conditions common in sewer networks.

2.5 Robotic Autonomous Control

Robotic autonomous control plays a critical role in ensuring the efficient operation of robots within underground pipe networks. These control systems enable robots to autonomously navigate, inspect, and maintain these environments with minimal human intervention. This section explores the key aspects of robotic autonomous control, focusing on control algorithms, decision-making processes, and adaptive strategies that are essential for autonomous operations within these often unmapped networks.

The complexity of underground pipe networks, particularly in water and buried pipe networks, requires robots to operate independently and adaptively (Wong et al., 2018). Various control algorithms have been developed to address these challenges. Mounce et al. discuss several critical algorithms, such as Lagrangian-based sensing, which allows robots to dynamically move within pipes while continuously collecting data (Mounce et al., 2021). This technique enhances data acquisition compared to fixed-point methods by enabling real-time environmental sensing.

Another key algorithm is Simultaneous Localization and Mapping (SLAM), which provides robots with the ability to create accurate 3D maps of the networks (Mur-Artal et al., 2015). SLAM is particularly useful in feature-sparse environments, where real-time positioning updates are crucial for navigation. Additionally, swarm robotics, inspired by natural systems like ant colonies, allows multiple robots to collaboratively inspect and monitor large sections of pipe networks (Jahanshahi et al., 2017). These robots share data and optimize their coverage without relying on centralized control, enabling more efficient and comprehensive monitoring.

Furthermore, AI and machine learning algorithms process the vast amounts of data collected by these robots, providing insights into pipe conditions and predicting potential failures (Wang et al., 2021). This enables autonomous decision-making, allowing the robots to proactively respond to changing environments. Adaptive control and motion planning algorithms also play a crucial role by enabling the robots to navigate non-static environments, avoid obstacles, and optimize energy use, ensuring safe and efficient movement through these networks.

For smaller robotic systems navigating in pipe networks, Nguyen et al. present a control algorithm based on a finite state machine (Nguyen et al., 2022). This system defines 13 distinct states, such as moving straight, turning, or dealing with dead ends. The robot uses sensor data to determine its current state and activate a corresponding decision-making protocol, allowing it to operate autonomously without external input.

The control strategy is divided into high-level and low-level controls. High-level control manages decision-making in pipe geometries, while low-level control focuses on maintaining the robot's stability and adjusting its speed. Navigation is supported by range sensors and an Inertial Measurement Unit (IMU), which eliminates the need for visual input and reduces computational demands. This system ensures that the robots can explore pipe networks comprehensively and autonomously, ensuring operational safety and thorough coverage.

For autonomous navigation, stable tracking control methods are essential for ensuring that robots can follow predefined paths accurately. Kanayama et al. developed a control system based on a Liapunov function that ensures stability during path-following in a 2D plane (Kanayama et al., 1990). The robot's position and orientation are monitored using a global Cartesian coordinate system, and deviations from the desired trajectory are corrected by adjusting the robot's linear and rotational velocities.

This method has been successfully implemented on real-world robots, such as the Yamabico-11, showing robustness in both simulations and practical applications. Stability is validated through the Liapunov function, which guarantees that the robot consistently returns to its intended path even in the presence of environmental disturbances. Velocity and acceleration limits further enhance the system's reliability by preventing slippage and maintaining control during sharp turns or non-smooth paths.

To enhance the adaptability and optimization of robot control systems, reinforcement learning (RL) algorithms have been applied to both single-agent and multi-agent systems. Kiumarsi et al. provide a comprehensive survey of RL algorithms designed for optimal and autonomous control (Kiumarsi et al., 2017). These algorithms focus on maximizing long-term rewards by learning from the environment, particularly in uncertain and changing conditions.

The On-Policy Integral Reinforcement Learning (IRL) Algorithm allows for real-time updates to the value function and control policy without requiring full knowledge of the system's dynamics. By iteratively improving through actor-critic methods, this algorithm ensures that robots achieve optimal control in challenging settings. H-infinity (H_∞) control algorithms, which treat control problems as zero-sum games between the controller and environmental disturbances, have also been employed to handle uncertainties and ensure stable performance.

Finally, OpenRAVE, developed by Diankov and Kuffner, introduces a modular approach to autonomous control with its Sense-Plan-Act loop (Diankov and Kuffner, 2008). This architecture integrates sensing, planning, and acting components, enabling robots to continuously collect data, generate navigation plans, and execute tasks while responding dynamically to environmental changes. The modularity of OpenRAVE allows it to be customized for various robotic tasks, making it highly adaptable for underground pipe inspection.

The system supports real-time feedback, enabling adaptive responses and flexibility during operations. By integrating with other frameworks, such as ROS, OpenRAVE facilitates smooth transitions between simulation and real-world execution, further enhancing the deployment of autonomous systems for challenging tasks.

The integration of control algorithms, decision-making processes, and adaptive strategies is crucial for the successful deployment of autonomous robots in underground pipe networks. These systems enable robots to navigate challenging environments, perform inspections, and carry out maintenance tasks with minimal human intervention. Projects such as Pipebots exemplify the effectiveness of these technologies in addressing the challenges of managing buried infrastructure (Team, 2019). By leveraging these cutting-edge technologies, autonomous robots can revolutionize the inspection, monitoring, and maintenance of underground pipe networks, significantly improving the efficiency and reliability of buried pipe network management.

In conclusion, while the primary localization techniques for robotic systems can be categorized into probabilistic filtering methods, SLAM techniques, landmark-based localization, and sensor-based approaches, each has its limitations. Probabilistic methods such as Kalman Filters and Particle Filters provide a strong foundation for estimating position in changing environments; however, they often struggle with cumulative errors over time, particularly during extended or challenging navigation tasks. SLAM techniques enhance a robot's ability to map and localize itself concurrently, though these methods tend to be computationally intensive and may falter in environments lacking distinctive visual features, such as subterranean pipe networks. Landmark-based methods offer adaptability via feature selection, but their dependence on identifiable landmarks becomes impractical in featureless or highly repetitive settings like pipes. Sensor-based approaches, including Radio SLAM, provide alternatives when conventional methods fail, but are vulnerable to signal interference and may prove less reliable in intricate underground contexts.

Topological mapping methods typically focus on the connectivity of the environment rather than precise geometric details. While graph-based topological maps effectively abstract challenging environments, such as underground pipe networks, they often fall short when precise navigation and mapping are required in environments with uniform, repetitive features, like cylindrical pipes. The integration of more geometric data can mitigate some of these shortcomings, but such hybrid approaches increase the computational load and are susceptible to environmental distortions. Furthermore, conventional topological methods may struggle with real-time processing demands, limiting their practical application in fast-moving autonomous systems.

This thesis presents a method that integrates precise geometric relationships into topological maps, thereby creating a hybrid model that combines the strengths of both topological and metric maps. While this hybrid approach integrates geometric relationships into topological maps, providing more accurate navigation, path planning, and localization, it is important to note that traditional topological methods have their own advantages, particularly in terms of computational efficiency. Topological maps, by focusing on the connectivity of the environment, allow for faster processing, especially in environments with repetitive structures, such as pipe networks. However, the addition of geometric data increases the computational load, which may present challenges in real-time systems. In comparison, existing works such as (Zhang et al., 2023) and (Blochliger et al., 2018) have shown that while topological maps are efficient in large-scale environments, they often struggle with precision in featureless or regular environments. This work aims to address these limitations by providing a balance between topology and metric information, although the increased computational complexity should be considered in environments where computational resources are limited. The enhanced topological mapping, combined with sensor fusion techniques, merges data from various sensors, including cameras and inertial measurement units (IMUs). This enables the generation of a detailed and accurate representation of the environment, ensuring robust performance even under challenging conditions such as poor lighting or obstructed views in underground pipe networks. This method addresses the shortcomings of conventional topological mapping and localization techniques, enhancing the robot's operational efficiency and reliability in confined environments.

Chapter 3

Real-time robot topological localization and mapping with limited visual sampling in simulated buried pipe networks

3.1 Introduction

The development of robot localization technology has gained significant attention due to its potential for improving the efficiency of critical infrastructure maintenance, such as pipelines (Jensfelt, 2001). However, pipeline robotics faces unique challenges compared to indoor environments, particularly in miniaturizing robots for autonomy. These robots require accurate sensors, efficient power sources, and real-time operational algorithms, while also ensuring computational efficiency to minimize power and hardware requirements. Traditional infrastructure maps often fail to represent the current state of pipeline networks, leading to outdated information that can hinder maintenance and repairs (Mounce et al., 2021). Autonomous mapping techniques, which provide real-time updates, offer a solution to this issue. However, for efficient navigation, traditional geometric maps may not capture the full connectivity and relationships of pipeline components, such as junctions. This limitation calls for topological maps, which focus on the connectivity and spatial relationships within the network. These maps are more computationally efficient, reduce error accumulation, and allow for advanced reasoning and planning (Kortenkamp and Weymouth, 1994).

This work proposes a topological mapping algorithm for autonomous navigation in buried pipe networks. Topological representations are compact and intuitive, making them well-suited for real-time applications in constrained environments. By leveraging these maps, robots can efficiently navigate and perform inspection tasks in dynamic, underground settings. A commonly used technique for mapping and localization in such environments is Simultaneous Localization and Mapping (SLAM), which typically uses geometric representations (Mur-Artal et al., 2015; Rugg-Gunn and Aitken, 2022). However, our approach focuses on topological maps, which are more efficient in terms of storage and computation, offering a robust solution for autonomous navigation over extended periods.

We present a hybrid approach integrating vision- and nonvision-based strategies in a single fully autonomous framework. In particular, we use non-vision-based exploration and navigation while our localization and mapping rely on an active-vision algorithm, which is only activated at topologically key (landmark) locations. Nguyen et al. (2022) describe an autonomous robot for pipeline inspection which they use to demonstrate exhaustive non-visionbased exploration of a physical pipe network. The small size of the robot (70 g, navigating through 150 mm pipes) imposes severe limitations on sensors, motors and battery power. Instead of using a camera, the robot uses distance sensors to navigate through the pipes and to execute turns and maneuvers at junctions and dead ends. Here, we propose to add localization and mapping capabilities to this platform. We implement this robot in a simulated pipe network to develop and test our active vision, localization and mapping algorithm. In our work, we focus primarily on junctions, as these are sufficient for the topological mapping of the network. Once the robot arrives at a junction, it uses its camera to collect an image dataset of the location. The robot localizes by comparing the images from its current location with the data in its database. This localization step includes orientation matching, i.e., calculating the robot's rotation relative to the orientation in the database. An active vision step is included to increase the accuracy and robustness of the localization. Finally, the robot either identifies its location based on a good match with an existing location in the database or defines its location as novel, adding the location information and the associated image set to the database and the topological map.

This chapter is organized as follows. In addition to the creation of topological maps such as mentioned above, we have also developed control algorithms for the fully autonomous navigation of the robot in our experiments, so that the robot can generate topological maps in real-time as it explores the pipeline. The methods are described in Section 3.2, including a brief overview of the autonomous control of the simulation robot presented by Nguyen et al. (2022) (Section 3.2.1). Although the physical robot was successful in exploring the pipe network, this study relies on a motorized camera that has yet to be implemented. Hence, the experiments with localization and topological mapping are conducted in a simulated environment. The results are presented in Section 3.3, including simulations of the robot recognizing junctions in real-time and constructing a topological map as it moves through the pipeline (Section 3.3.6). Finally, the chapter concludes with a discussion in Section 3.4.

3.2 Materials and methods

The algorithm for automatic control is provided by Nguyen et al. (2022), and a detailed method overview can be found there. We present a brief discussion in Section 3.2.1. The experiments in this study were conducted within a small-scale pipe network simulated in the Gazebo simulator. The network consists of six junctions and relatively simple pipe structures (Figure 3.1). Additionally, the interior of the pipes was textured, which might influence the image matching accuracy. The limited size of the network and the controlled nature of the simulation environment should be noted as these factors may not fully capture the complexity of real-world applications. The junction recognition process involves distinguishing between known and unknown junctions (Section 3.2.2). The image matching used for calculating the similarity between junctions relies on NCC (Section 3.2.3). By comparing the current junction with all the junction data collected in the database, the robot's task is to determine whether the junction has been visited before. If it has, the robot proceeds to localize (Section 3.2.4). If the junction is new, it is added to the image database and the topological map is updated (Section 3.2.5).

3.2.1 Robot control for navigation and exploration

The miniature robot is equipped with three range sensors, an inertia measurement unit (IMU), two wheel-leg encoders, and a camera (800 × 800 Pixels) for sensing. The robot starts at the entrance of the pipe network and exhaustively explores all the sections before returning to the starting point. While exploring the pipe network, the robot encounters all junctions, dead-ends, and obstacles at least once, using its sensors to autonomously navigate and maneuver to deal with these conditions based on its estimated states. In our work and Nguyen et al. (2022), we do not use the camera for autonomous control of the robot but only fuse the data of the other three types of sensors to estimate the robot's state. Eleven robot-in-pipe states capture different positions at T-junction, at left/right branches, at left/right corners, at straight pipe centerline, inclined in straight pipe, at a dead-end, at a cross junction, at open-space, and finally, to detect when the robot is approaching a collision. Three time-of-flight range sensors are installed at the front, front-left, and front-right of the robot to measure distances from the robot to the surrounding environment. These distances, combined with IMU data and their historical data, provide sufficient information for the algorithm to calculate the current robot's local state (Nguyen et al., 2022).

Once the robot confirms its estimated local state, it makes a high-level decision to turn right or left at an angle, go straight, or turn around. By default, the robot decides to always take the right most unexplored branch at any junction to guarantee an exhaustive exploration of all sections of the pipe network. Specifically, the robot turns right at a cross junction, T-junction, right corner, and right branch. It goes straight at a left branch and in a straight pipe. It turns left at a left corner and turns around at a dead-end or upon encountering a significant obstacle (that blocks passage through the pipe cross-section). Depending on the direction of an impending collision, the robot turns left or right to move away from the pipe walls or obstacles. A detailed table of actions taken regarding the robot state was explained in Nguyen et al. (2022). After making a high-level decision, a low-level motor controller is called to fine-tune the robot's direction and velocity in the pipes. Encoder values and their historical data are used to calculate the maneuver's speed and estimated turning angles.

As described above, the robot uses three distance sensors mounted on the front to recognize a junction ahead. At this point, the robot faces the center of the junction ahead. In our work, we augment the control algorithm in Nguyen et al. (2022) to command the robot to move forward from the entrance to a junction by a set distance (equal to the estimated radius of the junction). Once at the approximate center of a junction, sampling, localization, and mapping take place, as described in the following sections. Once complete, the robot returns to the entrance of the junction, and the original control algorithm resumes.

We noted that the IMU, odometry and laser data are not used for localization but only for navigation and low-level control. Our localization method as described in this chapter relies on vision information only.

3.2.2 Identification of known/unknown junctions

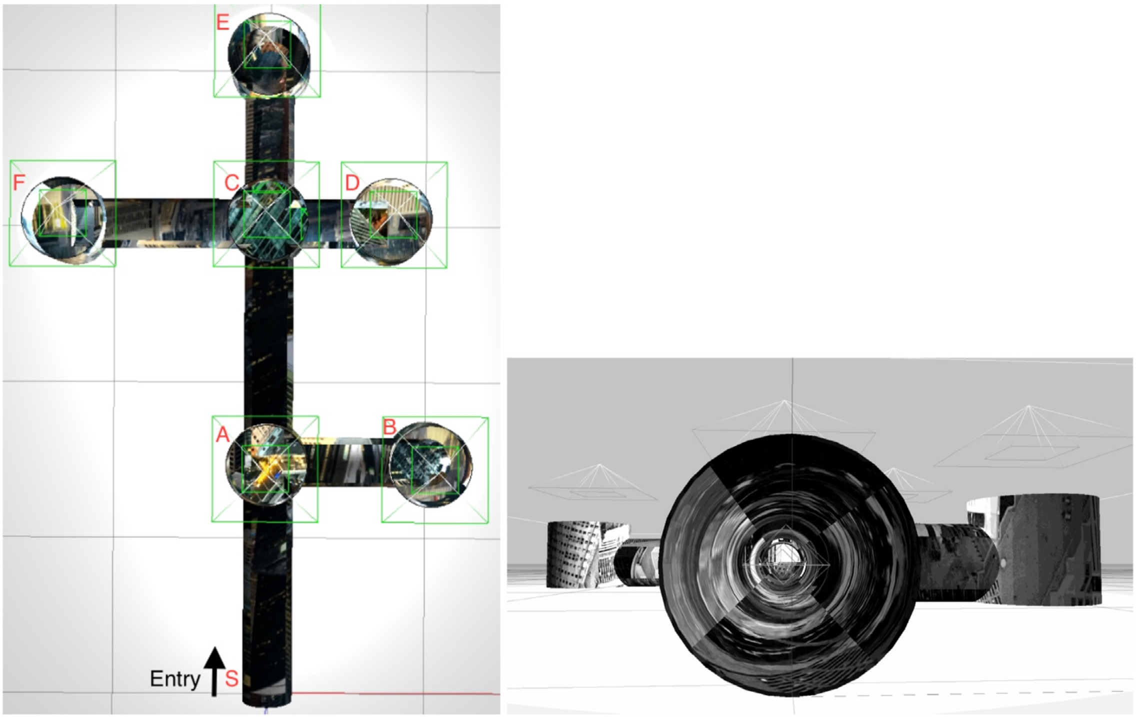


Figure 3.1: The simulated pipe network in the Gazebo simulator (Koenig and Howard, 2004). Pipe diameter 200 mm. Left: External view with the six junctions and entry point manually labeled. Right: inside view of a pipe with added texture.

The robot visits different junctions as it explores the pipe network (Figure 3.1). During this exploration, the robot may visit the same junction multiple times via different paths. In a tree-like network structure, the robot is guaranteed not to encounter circular paths and can always return to the starting point. We call previously visited junctions known junctions. Junctions encountered for the first time are referred to as unknown junctions. This section describes how the robot can identify whether it is currently in a known or unknown junction. The robot identifies junctions using a NCC-based image method, which doesn't require knowing the specific path taken to reach the junction. As the robot autonomously explores and performs topological mapping, the database consists only of known junctions. As more junctions are encountered, the database and map are updated.

When the robot is approximately at the center of a junction (Figure 3.2), it collects images all around the junction (See example in Figure 3.3). The 360° view

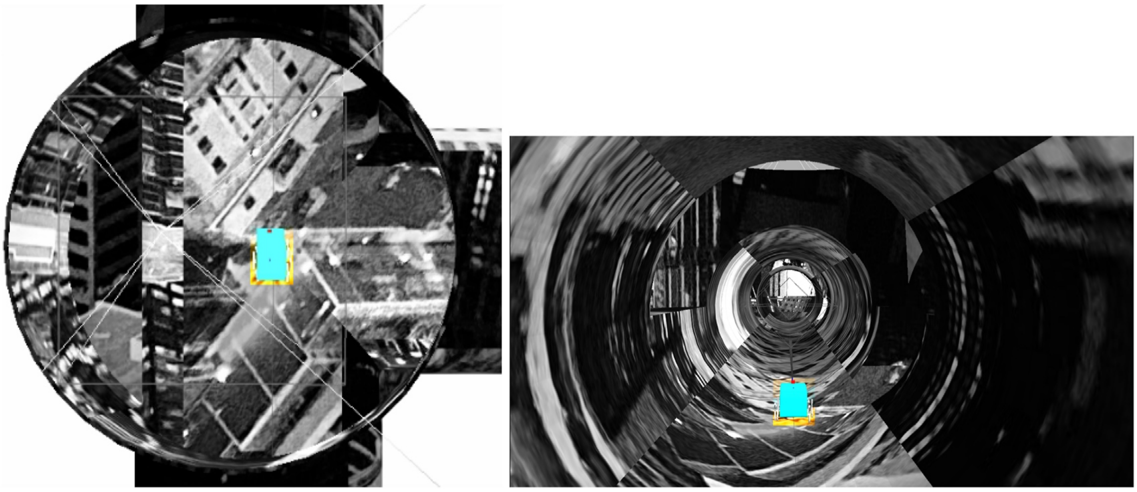
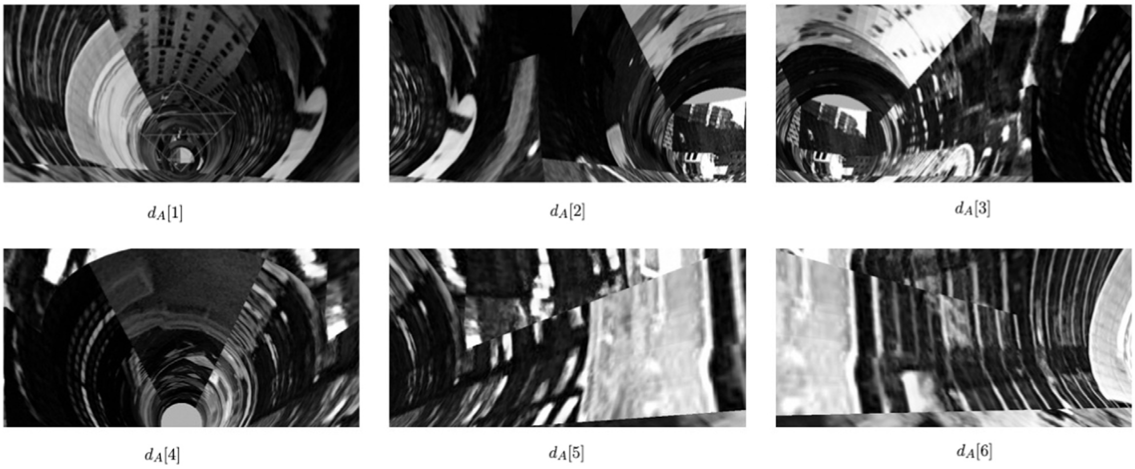


Figure 3.2: Robot (appearing as a light blue rectangle in a top-down and rear view) at the approximate center of junction A.



$d_A[1]$ $d_A[2]$ $d_A[3]$

$d_A[4]$ $d_A[5]$ $d_A[6]$

Figure 3.3: Collected image set from junction A (example of d_A).

of the current junction is obtained by rotating the camera that is mounted above the robot (Figure 3.5) in discrete steps. Given the diameter of the simulated pipe, each image covers an angular width of about 90°. We want a significant overlap between images and found that a small set of 6 images (corresponding to 60-degree rotations between images) is sufficient for robust localization (see Section 3.3.2 in the Results). As the robot explores the environment, it collects images from each junction, and the stored sets of images are given by:

$$D = \{d_i\}_{i=1}^{\text{NumJunctions}}$$

where D is the database, and NumJunctions is the number of known junctions at that point in the exploration.

To identify a junction, the robot compares the image set collected at the current junction, d_c , with the image sets of all known junctions. Given two image sets d_c and d_i, we perform image matching to define the similarity $S(d_c, d_i)$ between the two junctions (see Algorithm 2 below). We then compute the maximum similarity score, S_{max}, across all possible known junctions, to obtain the best candidate for a known junction, here labeled with index j, using:

$$(S_{\max}, j) = \Big(\max(S(d_c, d_i)), \arg\max_{S}\Big)$$

Finally, we determine whether the current junction is a known junction or an unknown junction by thresholding this maximum similarity score S_{max}, shown in Algorithm 1. If the similarity is higher than the threshold, we identify the current location as a known junction j in the database. Otherwise, we perform active vision $A(d_c, d_j)$ to better align the image sets (see Section 3.2.4; Algorithm 3) and repeat the image-set comparison with junction j in the database. If the similarity is still lower than the threshold, we define this junction as an unknown junction and add the current image set d_c to the database as a new junction. We also add this junction to the topological map, as described in Section 3.2.5.

3.2.3 NCC-Based Image Matching and Preprocessing

Normalized Cross-Correlation (NCC) is a widely adopted method in image matching due to its effectiveness in quantifying the similarity between two image regions. The rationale for using NCC in this study stems from its ability to measure the similarity of corresponding pixel intensities between images, independent of their overall intensity or brightness. This makes NCC suitable for matching images in grayscale, where variations in lighting conditions and exposure are not significant factors (Zhao et al., 2006).

The NCC method computes the correlation coefficient between the target image and the reference image, measuring the alignment and matching of visual features. In the context of this research, the NCC-based image matching approach is used to identify the most likely corresponding junction in the database based on the robot's current visual input. The algorithm is designed to compute the similarity between a set of images from the current location of the robot and images of candidate junctions, ensuring that even slight variations in perspective or orientation are accounted for through pixel shifts (Rao et al., 2014).

Algorithm 1: Algorithm for identification of known/unknown junctions.

```
Knoen/Unknown junction $K(d_c, d_j)$;
Input  : Current image set $d_c$, corresponding max similarity image set $d_j$
         in the database
Output: known/unknown junction $True/False$
if $S_{max} > similarity\_threshold$ then
    return $known$;
else
    $S_{max} = A(d_c, d_j)$;
    if $S_{max} > similarity\_threshold$ then
        return $known$;
    else
        return $unknown$;
    end
    return $unknown$;
end
```

While NCC is robust and simple, it has its limitations, especially when the images contain significant variations in scale, rotation, or affine transformation. Moreover, NCC is sensitive to noise and may not perform well in the presence of dynamic changes in the environment, such as occlusions or moving objects. For more complex scenarios, alternative methods like Scale-Invariant Feature Transform (SIFT) (Lowe, 2004), Speeded-Up Robust Features (SURF) (Bay et al., 2006), or Oriented FAST and Rotated BRIEF (ORB) (Rublee et al., 2011) could be considered, as they are designed to handle scale and rotation-invariant matching. However, we also tried feature-based matching algorithms like SIFT, SURF, and ORB, but in the pipeline environment, these methods failed to detect sufficient features, and their performance was poor.

Thus, NCC was chosen in this study for its simplicity, efficiency, and suitability for comparing grayscale images of similar content. The limitations of NCC can be mitigated by employing additional pre-processing steps, such as cropping the lower portion of the image that might be dominated by the robot's body, as described earlier. This ensures that only relevant features of the junction are considered during the matching process, thereby increasing the robustness of the NCC-based approach (Parker, 2010).

In summary, while NCC provides an efficient and straightforward method for image matching in our scenario, it is important to consider its limitations in real-world applications, where more advanced techniques like feature-based methods might be necessary for handling more complex visual matching tasks. Nevertheless, within the scope of this study, NCC remains a suitable approach for matching image sets, as it efficiently calculates the maximum similarity and corresponding offsets between images. The following sections describe the details of how NCC is applied in our approach to image matching and its preprocessing steps.

NCC is used to calculate the maximum similarity, S(d_c, d_i), and corresponding offset between two different image sets. Let W_1 and W_2 be two matching windows of the same size $m \times n$ within images I_1 and I_2, respectively, with image dimensions $M \times N$. The average grayscale values of the matching windows are denoted as u_1 and u_2, where:

$$u_1 = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} W_1(x+i, y+j), \quad u_2 = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} W_2(x+i, y+j).$$

The normalized cross-correlation (NCC) between the two windows W_1 and W_2 at position (x, y) is given by:

$$NCC = \frac{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(W_1(x+i,y+j)-u_1)(W_2(x+i,y+j)-u_2)}{\sqrt{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(W_1(x+i,y+j)-u_1)^2}\sqrt{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(W_2(x+i,y+j)-u_2)^2}}.$$

For greater realism, only grayscale information is used for image analysis. Image matching is performed for every possible junction pairing (one corresponding to the robot's present location and the other for each 'target' junction in the database). There are six images in each image set, and hence six possible offsets between two image sets. For a given offset, the normalized crosscorrelation is computed (in the Fourier domain) between all images in the current location with the candidate matching junction in the database. For each candidate junction, after finding the image offset with the maximum similarity, the horizontal pixel offset is set to the position of the maximum cross-correlation value. Looping through all possible candidate junctions, the pairwise similarity and the corresponding image and pixel shifts(offsets) are obtained. Finally, the highest similarity score is used to pin down a single candidate junction. The image-matching data for this junction pair is then used as a basis for the active vision step.

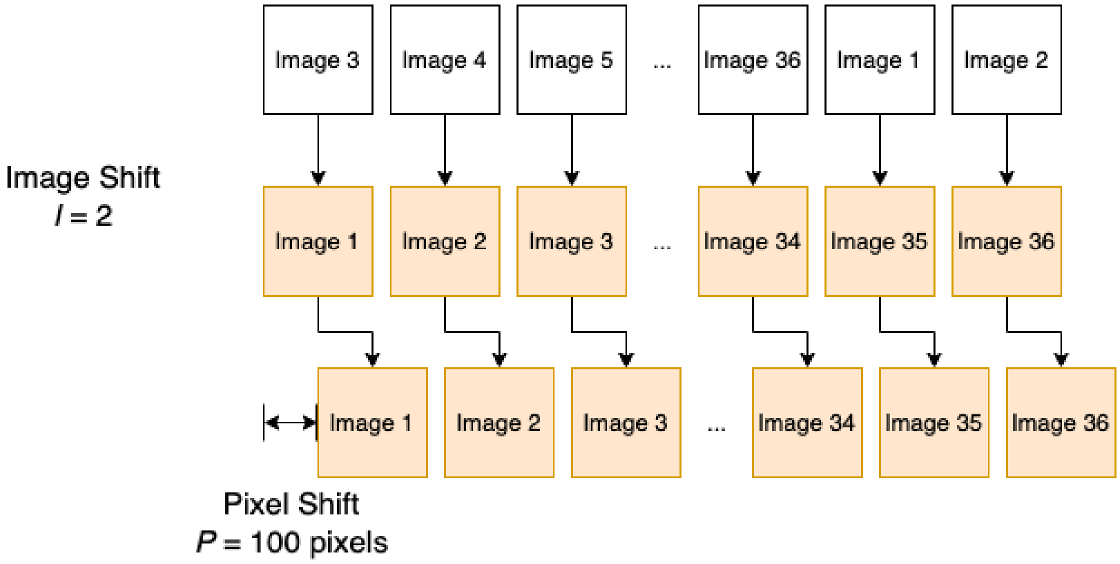


Figure 3.4: An illustration for the image shift I and pixel shift P. (white: currently sampled junction, orange: image set of a known junction from the stored dataset).

NCC is a commonly used region-based method in image feature matching. In the absence of a single 360° panoramic view of the entire junction, here we apply NCC to one image-pair at a time and increment the possible image shifts to allow for the robot entering the junction from different pipe sections, as illustrated in Figure 3.4. We match the similarity of gray pixel values in the whole image field in the two images (I_1 and I_2). For I_1, we calculate the NCC coefficient between it and the I_2, and the corresponding pixel shift that maximizes the match.

We exploit the similarity in the formulation of the crosscorrelation to a convolution function, which can be implemented efficiently with a Fast Fourier Transform (FFT). To perform a crosscorrelation, we perform a convolution between one image and a second conjugate image (see Algorithm 2). The implementation uses python3.6, opencv2, Numpy, and fftconvolve function from scipy.signal.

We note that a preprocessing step is taken before performing the cross-correlation. Because the lower part of the image contains the robot, which will affect the accuracy of the comparison results, we crop out the lower half of the image. Next, we subtract the mean pixel intensity.

3.2.4 Pipeline robot localization using active vision

As previously mentioned, a minimal change to this control algorithm was implemented to allow the robot to approach the approximate center of the junction. In this section, we describe a second addition to the control algorithm: the implementation of active vision.

Given the image shift I, pixel shift P, as well as the angle between successive images ($2\pi/6$ radians), camera width ($2\pi/4.5$), and pixel widths (800 pixels), the relative angle between the two image sets, is ($I * 2\pi/6 + P * cam_width/pixel_width$). To execute the active vision, we first instruct the robot to move forward by a pre-estimated distance. At this point, the robot is already near the center of the junction, and this movement serves only as a fine adjustment. The robot then moves slightly forward, backward, left, and right, with each movement approximately equal to the diameter of one wheel, to refine its position and maximize the image similarity score through active vision. We continue cross-correlating with the same dataset in the database until we find a direction that increases the similarity. If active vision increases the similarity sufficiently to exceed to similarity threshold, the junction can be identified (it is a known junction). Otherwise, it is classified as an unknown (i.e., new) junction, as described in Section 3.2.2; (Algorithm 3).

3.2.5 Topological mapping

Topological maps capture the relationships between elements of a map that are represented as nodes (or vertices), connected by edges (Garcia-Fidalgo and Ortiz, 2015). In our case, the map captures connectivity between pipe segments and the correct order of these edges in each junction. While distance information can be important for some purposes, the adjacency of pipe segments and junction information suffice for path planning in the network. The key to constructing topological maps is identifying appropriate spatial points of interest in the environment as

Algorithm 2: Algorithm for computing the similarity of two image sets and corresponding image shifts, pixel shifts.

```
Image Set Cross Correlation $S(d_a, d_b)$;
Input  : Two Image Dataset $d_a$ and $d_b$
Output: Similarity $S$, Image Shifts $I$, Pixel Shifts $P$
/* Iterate through all the images and cycle through different
   image shifts                                              */
for $j \leftarrow 0$ to sizeof $(d_a)$ $J$ do
    for $k \leftarrow 0$ to sizeof$(d_b)$ $K$ do
        /* Read image from junction dataset a and b          */
        /* Pre-processing, read the image in grey channel and
           crop the bottom half                              */
        Image1 $\leftarrow d_a[j]$;
        Image2 $\leftarrow d_b[(j + k)\%K]$;
        /* Start Cross-Correlation between two images        */
        /* Calculate the difference between each pixel and the
           mean of every pixel of one image                  */
        Image1 $\leftarrow$ (Image1 - Mean(Image1));
        Image2 $\leftarrow$ (Image2 - Mean(Image2));
        /* Faster to flip up down and left right, then use
           fftconvolve                                       */
        Flip1 $\leftarrow$ FlipUpDown(FlipLeftRight(Image1));
        OutMatrix $\leftarrow$ fftconvolve(Image2, Flip1);
        /* Finally, normalize the resulting matrix           */
        similarity $s \leftarrow$ Max value in OutMatrix;
        pixel shifts $p \leftarrow$ Index of $s$;
        $s_{list} \leftarrow s$;
        $p_{list} \leftarrow p$;
    end
    /* Sum $s$ and calculate the mean                        */
    $s$ = Mean(Sum($s_{list}$));
    /* Sum $p$ together and calculate the mean               */
    $p$ = Mean(Sum($p_{list}$));
end
/* Find the max $s$ and $p$ based on different $j$ for a single
   candidate junction                                        */
$S \leftarrow s$;
$P \leftarrow p$;
$I \leftarrow j$;
return $S, P, I$;
```

Algorithm 3: Algorithm for using the active vision to find the maximum similarity of two image sets based on image shifts, pixel shifts.

```
Active Vision A($d_c, d_j$);
Input  : Current image set $d_c$, corresponding max similarity image set $d_j$
Output: Similarity $S$
/* Using Algorithm 2 to calculate the similarity $S$, image
    shifts $I$, pixel shifts $P$                                   */
S, P, I ← S($d_c$, $d_i$);
RelativeAngle ← ($I \times 2\pi/6 + P \times cam\_width/pixel\_width$);
Rotate the camera according to RelativeAngle;
/* Initialize $S_1$                                                */
/* Move the robot a little distance forward                        */
MoveRobot();
$d_c$ ← RecordImageSet();
$S_1$ = S($d_c$, $d_j$);
if $S_1 > S$ then
    return $S_1$;
else
    MoveBackCenter();
end
/* Move the robot a little distance backward                       */
MoveRobot();
$d_c$ ← RecordImageSet();
$S_1$ = S($d_c$, $d_j$);
if $S_1 > S$ then
    return $S_1$;
else
    MoveBackCenter();
end
/* Move the robot a little distance left                           */
MoveRobot();
$d_c$ ← RecordImageSet();
$S_1$ = S($d_c$, $d_j$);
if $S_1 > S$ then
    return $S_1$;
else
    MoveBackCenter();
end
/* Move the robot a little distance right                          */
MoveRobot();
$d_c$ ← RecordImageSet();
$S_1$ = S($d_c$, $d_j$);
if $S_1 > S$ then
    return $S_1$;
return $S$;
```

vertices and extracting sufficient spatial semantic information from these locations. In addition to their compactness and elegance, topological maps may be more stable than spatial maps in the face of closed loops (in the absence of absolute position information). However, our algorithm is specifically designed for tree-structured networks because the autonomous control strategy ensures exhaustive exploration only in loop-free environments. If loops were present, the robot could become trapped in a cycle, continuously revisiting the same paths without a reliable way to exit. Therefore, while topological maps could theoretically handle loops with additional mechanisms such as loop closure detection, our current implementation is limited to tree-structured networks to guarantee complete exploration and successful mapping. It is possible to consider a more spatially grounded topological map, in which edges (i.e., pipe sections) are assigned information (e.g., a distance estimate from dead reckoning, the time taken to move between two vertices, or the energy consumed). In our case, however, junctions contain sufficient semantic information about the scene.

Whether a junction is known or unknown dictates whether the topological map is up to date. Given an unknown junction, the robot firstly adds the image set d_c to the database D, creates a new vertex V_n, and adds an edge E between it and the previous vertex the robot visited V_{last}. When moving between two known junctions, the edge may or may not have been traversed previously. In the absence of an edge in the map, a new edge will be added between the previous junction to the current position. Topological maps typically encode vertices and edges in an adjacency graph. For navigation purposes, it is also useful to disambiguate the ordering of the edges for each junction. Here, we use the order in which edges are traversed to assign them integer labels (W). Whenever a new junction is added, if the edge used to reach the junction is new as well, it is assigned the next unused integer as its label. Thus, once the map is complete, each edge has a unique label (see Algorithm 4). Given the exploration algorithm (in our case, consistently taking the right-most branch), it is then possible to determine the relative orientation of the pipe sections.

3.3 Results

In this section, we present our experiment with a simulated robot in a pipe environment. Our overall goal is to test the performance of our system by building a topological map doing autonomous control. However, before we present these topological map-building results in Section 3.3.6, we will present a set of simpler experiments that we have performed to test the localization performance of our robot. The following five experiments will be presented.

1. *Perfect Center*: Our first experiment tests if the robot can identify which junction it is in with respect to the junction in the database when the robot is initialized at the center of the junction. In this experimental setup, we assume a precise center position for the junction. However, in real pipe environments, the existence of a well-defined center may not always be the case. Regardless, as long as the robot can return to a position close to the first arrival point,

Algorithm 4: Algorithm for processing a junction during topological map building.

```
Topological Mapping $T(d_c, D, V_{last}, W, G)$;
Input  : Current image set $d_c$, Database $D$, last vertex $V_{last}$, Label $W$, and
         Graph $G$
Output: Graph $G$
/* Iterate through all the image sets in the database $D$ */
/* Initialize $S_{max}, P_{max}, I_{max}$ */
for $i \leftarrow 0$ to sizeof (D) do
    /* Using Algorithm 2 to calculate the similarity $S$, image
       shifts $I$, pixel shifts $P$ */
    S, P, I $\leftarrow$ S($d_c, d_i$);
    if $S > S_{max}$ then
        $S_{max} = S$;
        $P_{max} = P$;
        $I_{max} = I$;
end
/* Find the max similarity $S_{max}$ through all the comparisons and
   corresponding $d_{max}$, image shifts $I_{max}$ and pixel shifts $P_{max}$
   */
/* Determines if the current junction is a known or unknown
   junction */
result = $K(d_c, d_{max})$;
if result == $True$ then
    /* It's a known junction.  The vertex is the same as the
       vertex represented by $d_{max}$ */
    $V_c \leftarrow$ GetVertex($d_{max}$);
    last vertex $V_{last}$;
    if Edge($V_{last}, V_c$) doesn't exists then
        /* Create a new Edge $E$ */
        E $\leftarrow$ NewEdge($V_{last}, V_c$);
        /* Add the label $W$ to the edge $E$ */
        AddLabel($E, W$);
        $W \leftarrow W + 1$;
        /* Add Edge $E$ to Graph $G$ */
        $G \leftarrow$ AddEdge(E, G);
        return G;
    else
        /* It's an unknown junction */
        /* Add the current image set $d_c$ to the database $D$ */
        $d_c \rightarrow D$;
        /* Add a new vertex $V_n$ */
        $V_n \leftarrow$ NewVertex($V_n, d_c$);
        G $\leftarrow$ AddVertex($V_n$, G);
        /* Create a new Edge $E$ */
        E $\leftarrow$ NewEdge($V_{last}, V_n$);
        /* Add the label W to the Edge E */
        AddLabel($E, W$);
        $W \leftarrow W + 1$;
        /* Add Edge E to Graph G */
        G $\leftarrow$ AddEdge(E, G);
        return G;
```

the method can still function effectively. In this experiment, autonomous navigation is not used. We also use a robot initialized at the center of the junction to determine how many images to collect in each junction. The experiment suggests that six images are sufficient (see Section 3.3.2).

2. *Noise*: To validate that six images suffice even for the imperfect positioning of the robot in the center of the junction, we repeated the above experiments such that robot will be located at 3 cm off center (see Section 3.3.3).

3. *Active*: To test the robot's ability to improve its localization through active vision, in this set of experiments, the robot is initialized at the off-center position and uses active vision (see Section 3.3.4).

4. *Entrance*: To more realistically simulate the process of the robot traveling autonomously in the pipeline, we place the robot on the edge of the junction in the pipeline, and perform a similar set of experiments. The robot moves forward a distance of the junction radius to the center before switching on its camera (see Section 3.3.5).

5. *Autonomous*: The robot is simulated under the fully autonomous exploration and control mode. A topological map is built (see Section 3.3.6).

3.3.1 Experimental setup

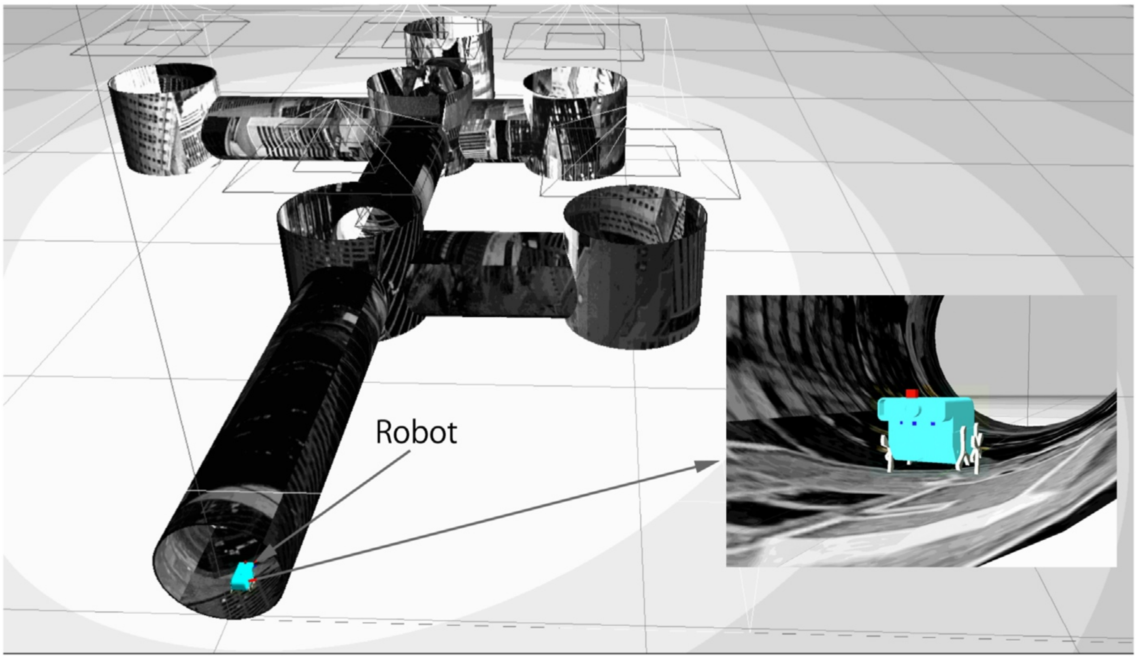


Figure 3.5: Robot simulated in the pipe network.

We model the mobile robot in a pipe network (Figure 3.1), and test our localization algorithm using ROS (Quigley et al., 2009) and the physics simulator Gazebo 9 (Koenig and Howard, 2004). The robot 3D model was designed in

Solidworks (see Figure 3.5; Nguyen et al. (2022) for details of the design) and imported into Gazebo. The imported model in Gazebo is shown in Figure 3.5. The robot has six wheel-legs, with wheel diameter 28 mm. Three left wheel-legs are connected and are actuated by a DC motor. Similarly, three right wheel-legs are connected and controlled by the second DC motor. The two motors are independent and controlled by two PID (proportional integral differential) controllers. Each motor controller is implemented in Gazebo-ROS using the ros_control package (allowing complex joint control algorithm to be applied to the DC motor instead of standard differential drive).

Figure 3.5 shows the simulated pipe structure in Gazebo and the simulated pipe network with the robot. The grayscale texture was imposed on the simulation environment to ensure that junctions look distinct. This is an underlying assumption of our work–i.e. that there is a visual difference between geometrically identical junctions. In a pipe network of any age, this is likely to be true but needs to be experimentally verified in future work. The robot is controlled by sending commands to its two ros_control motor controllers. The simulation provides an idealized scenario for the robot model, in which we can develop and test the proposed algorithms for localization and mapping.

3.3.2 Initializing the robot at the center of the junction

We collected images for each junction, sampled with a robot positioned precisely at the center of each junction as a reference database. Figure 3.3 shows sample data collected from one junction.

To determine a robust number of images that would suffice for localization, we performed a preliminary experiment, in which we placed the robot in the center of each of the six junctions in our network. As each image has a view of about 90°, more than four images would be required to ensure some overlap between adjacent images. We compared image sets with 6, 9, 18, and 36 images. For each case, four image sets were collected from each junction, facing four different directions (at 90-degree intervals) resulting in 24 image sets. We used the same robot to collect all the data sets required for the experiment at the same position in the pipeline. Defining the image sets from different robot orientations of the same junction as "Same" and image sets from different junctions as "Different". We calculated the similarity scores to the Same and Different junctions for image sets with 6, 9, 18, and 36 images. We present the results in Figure 3.6. While increasing the number of images yields higher accuracy, even with only six images, junctions can be recognized and the rotational error is small ($\leqslant 3°$). The result demonstrates the robustness of the algorithm to the number of images used, with six images being sufficient to reliably identify junctions under perfect center conditions.

To verify whether we can distinguish between known and unknown junctions, we collected sets of images for six junctions within the pipeline under different experimental conditions. For each experimental condition, we collected 5 imagesets at each junction giving us a total of 30 imagesets. We then compared the collected imagesets pairwise. Based on whether these sets of images originated from the same junctions or different junctions, we computed similarity results for junctions

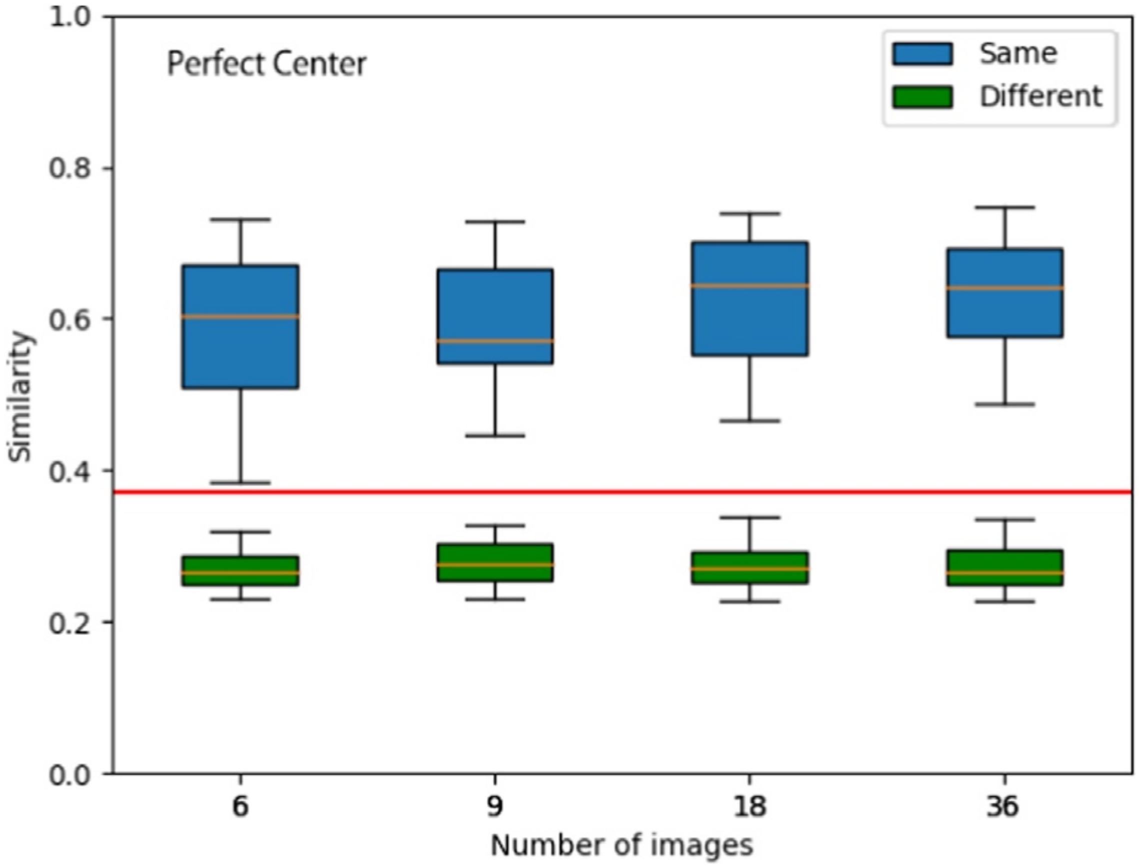


Figure 3.6: Assessing junction similarity for different sized image sets. Junction similarity results for 24 image sets (4 per junction at 90-degree intervals) under Perfect Center conditions. The similarity is compared for different numbers of images per image set. Boxplot (showing the Upper whisker, Upper quartile, Median, Lower quartile, and Lower whisker from the top to the bottom for the known junction (blue) and unknown junctions (green).) Applying a threshold of 0.36 (red horizontal line) would allow for accurate recognition of known junctions and distinguishing unknown junctions.

Table 3.1: Summary of similarity results across different image sets under different conditions. Note the low similarity obtained for unknown junctions, across all conditions.

Similarity ± std	Perfect center	Noise	Active	Entrance	Autonomous
Known junction	0.700 ± 0.157	0.614 ± 0.112	0.675 ± 0.166	0.643 ± 0.150	0.631 ± 0.109
Unknown junction	0.276 ± 0.028	0.276 ± 0.027	0.276 ± 0.024	0.280 ± 0.031	0.273 ± 0.025

Table 3.2: Summary of rotation error across different image sets under different conditions.

Rotation error (radians)	Perfect center	Noise	Active	Entrance	Autonomous
Known junction	0.054	0.044	0.051	0.047	0.101
Unknown junction	0.377	0.473	0.610	0.717	0.856

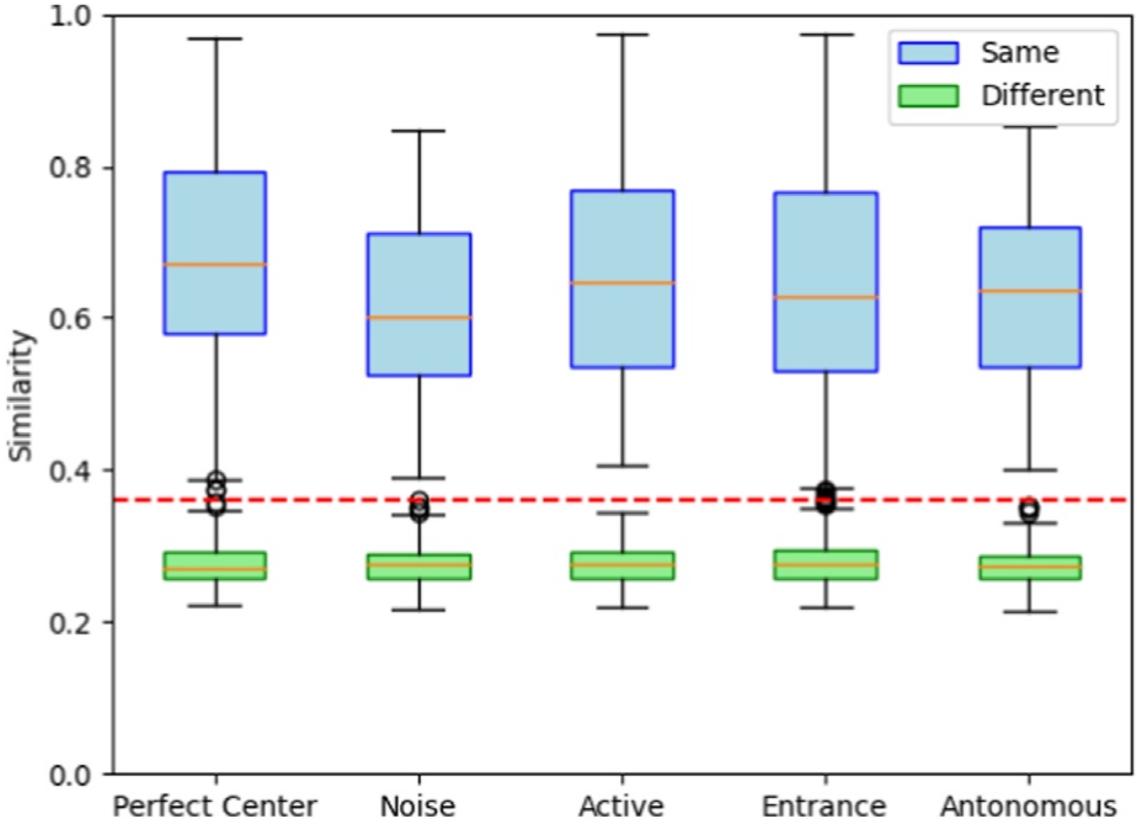


Figure 3.7: Image matching between junctions in the simulated pipe network (orange line: mean values, lower quartile: median of the lower half of the dataset, upper quartile: median of the upper half of the dataset, red line: threshold 0.36). Similarity scores for "Same" are computed as $S(d_{i^1}, d_{i^2})$, where d_{i^1} and d_{i^2} are different imagesets captured from the same junction i. Similarity scores for "Different" are computed as $S(d_i, d_j)$, where d_i and d_j are different imagesets captured from the different junctions, i and j. Same and different junctions can be robustly separate by the threshold.

labeled as “known” and “unknown”. Notably, each known category comprises 60 data points(imageset pairs coming from same junction) in its box plot, while the unknown category encompasses 375 data points(imageset pairs coming from different junctions). These results are presented in Tables 3.1, 3.2; Figure 3.7.

3.3.3 Initializing the robot in the off-center junction position

To validate that six images suffice even for the imperfect positioning of the robot in the center of the junction, we repeated the above experiment for eleven image sets collected from robots that were located 3 cm off the center from a random direction (‘Noise’) (Tables 3.1, 3.2; Figure 3.7). Henceforth, all image sets consist of six images per junction. Next, we tested the robustness of our junction identification algorithm. To verify that the identification of ‘known’ junctions was reliable, we also checked the relative angular offset (obtained from the image and pixel shifts) and confirmed that these agreed with the different robot orientations.

3.3.4 Using active vision

Our next experiment tests the robot’s ability to improve its localization through active vision. While our noisy experiment still showed good results, we expect that more realistic pipe conditions may result in greater slippage or possible errors in estimating the radius of a junction. To address such conditions, here, we move the robot’s position to bring it closer to the position at the time of image set acquisition in the database (corresponding to previous visits to the known junction). Active vision dramatically improves the robot’s position recognition accuracy over the ‘noise’ experiment above.

3.3.5 Initializing the robot at the entrance of a junction

After setting up a simulation environment, we applied the same control methods described in Nguyen et al. (2022) to the simulated mobile robot. We updated the algorithm so the robot could move to the center of the junction and execute image collection for localization and mapping. The robot was able to explore the simulated pipe network exhaustively in the simulation environment, but also experienced failure in some maneuvers due to sensitivity to positioning, when resuming navigation of the network after visiting a junction. Here, we focus on results from successful runs.

To more realistically simulate the process of the robot traveling autonomously in the pipeline, we place the robot on the edge of the junction in the pipeline (‘Entrance’). The robot is oriented toward the center of the junction where its distance sensors detect a junction ahead while advancing in the pipeline. From this ‘entrance’ point, the robot moves forward at a fixed distance, set to be the radius of the junction. As in previous experiments, we collect image data at the position where the robot finally stops, which can better simulate the heading error in the real world. We collected eleven image datasets of robots after having moved autonomously from the edge to the center of the junction. A comparison of these eleven image sets' data in the database is shown in Tables 3.1, 3.2; Figure 3.7.

From Table 3.1 results and using a threshold of 0.36 in the similarity, the robot can recognize the same junction, even off-center. However, it should be noted that this threshold may not be universally applicable and could depend on specific conditions such as the quality of the images, the camera's field of view, or the environment in which the robot operates. In our experiment, the 0.36 threshold was effective for the specific setup, but further tests under varying conditions would be needed to determine its robustness across different scenarios. In the third step of the active vision experiment, the robot achieves slightly improved mean similarities but is subject to larger variability. The robots from the junction entrance were all identified quite well and were considerably more accurate than the second and third sets of experiments.

3.3.6 Autonomously generating an ordered topological map

We now test our method under fully autonomous conditions. For this experiment, the junction database is initially empty and is updated every time the robot traverses a new and different junction, from those previously visited. We conducted 7 experiments each starting at different position in the pipe network shown in Figure 3.1. The starting positions were: S,A,B,C,D,E,F. The robot was able to generate the correct topological map successfully from all of these starting positions. In Figure 3.8, we shown one example. The robot starts at entrance location S and exhaustively explores the network until it returns to the entrance. At every junction, based on the image database of previously visited junctions and the robot's ability to recognize the current junction, the robot's topological map is updated in real-time. A successful map is shown based on the above perfect junction sampling.

We define the topological map as an undirected graph in which each node represents a junction that was visited and imaged by the robot. Each time the robot identifies its location with a junction in the database, it successfully proceeds without updating the map. Conversely, when the robot arrives at a new, unfamiliar node, the current junction and its set of images are successfully added to the database, a new node is added to the graph, and an appropriately labeled edge connects the previously visited node to the current node (Figure 3.8).

3.4 Conclusion

Our work employs a multimodal sensing strategy to combine a miniature autonomous robot's exploration, localization, and mapping in a simulated pipe network. While the results presented here are based on simulation experiments, we note that the robot is based on the SolidWorks model used to build a physical miniature model that successfully explored a similar pipe network. A key challenge in this work arises from the miniature robot's limited mobility and limited power. Hence we sought an efficient (powersaving) strategy that requires relatively little computation and storage. We rely on our finding in (Nguyen et al., 2022) that the range sensor data in the physical robot is accurate and

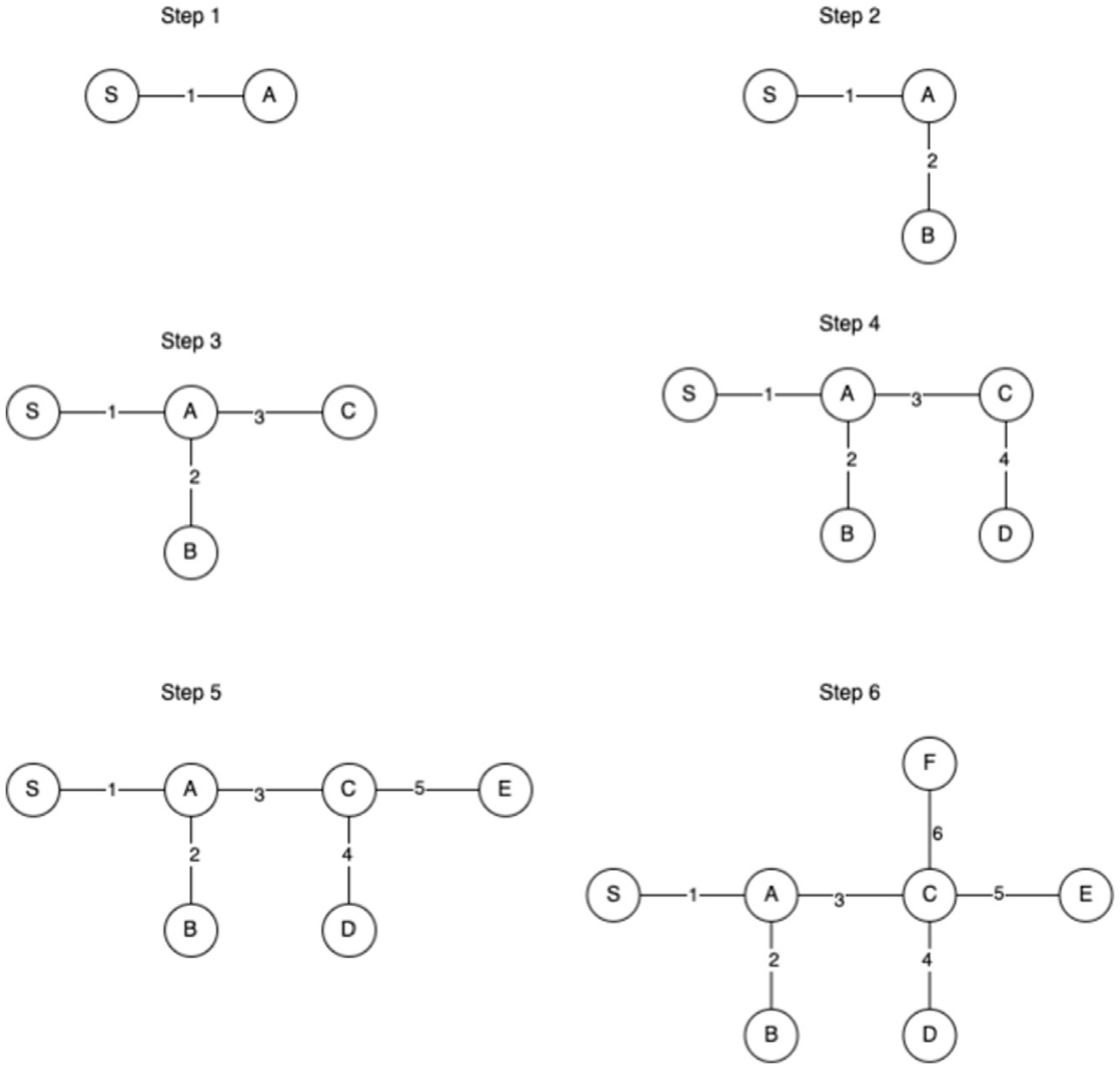


Figure 3.8: Dynamic topological map, generated during autonomous exploration of a simulated pipe network (shown in Figure 3.1). The six steps correspond to the six updates, each time a new (unknown) junction is visited. The edge labels represent the order in which the junctions were added to the map, and their relative orientations were determined with the knowledge of the rule-based exploration algorithm, yielding an ordering consistent with the pipe network. For visualisation, 90°angles were always assumed between adjacent edges since this is usually the case in real life situations.

can make good estimates of the surrounding environment when the robot executes navigation and obstacle avoidance. We proposed that vision sensors are rich in data and we demonstrate that using fast NCC methods suffice in our experiments to identify environmental landmarks. We adapted traditional image-matching techniques to pipeline geometries and rotational movement of the camera. We further propose that active vision may be useful to compensate for the limitations of poor positioning of robots during realistic autonomous movement in pipeline environments. It is important to highlight that although a specific robot model was used in our simulated experiments, the vision-based localization technique we propose is designed to be compatible with various robotic systems, rather than being limited to a particular robot. As long as a robot is capable of exploring the pipe network and equipped with a camera, our method can be employed for topological mapping of the network.

Our work primarily focuses on developing image-matching algorithms for active vision. However, it is important to address the main limitation encountered with the physical robot, which relates to the location and motorization of the mounted camera. In future work, we plan to conduct tests and evaluations in a physical setting using either the same robotic platform or an alternative one that allows for panoramic imaging, enabling us to assess the performance of both the vision-based algorithm and the control mechanism. It should be noted that the sensors used in our work were studied solely in the simulation environment, and therefore, it is challenging to reliably estimate the accuracy of the robot in an actual physical setting. Furthermore, conducting experiments with physical robots in realistic pipe networks introduces additional complexities due to sensor limitations, robot stability, and control issues. It is crucial to acknowledge the significant challenges associated with system integration and electronic isolation in real-world pipe networks, which are often wet, dirty, and cluttered. To thoroughly evaluate the performance, reliability, and durability of the in-pipe robot, extensive testing and validation in real-world pipeline environments are essential. Another noteworthy challenge in real-world deployments is failure recovery. For instance, if the robot becomes stuck inside the pipe network, our current method does not provide a solution for rescuing it. Additionally, realpipe networks often pose the challenge of dark lighting conditions. Consequently, in a practical deployment, the robot may require onboard lighting to support the camera's visibility. Under such settings, the usefulness and feasibility of active vision can be more extensively tested. The experiments conducted in our work have primarily focused on a specific pipe texture. Conducting experiments in a physical pipe environment will enable us to test our proposed method with real pipe textures and verify our hypothesis that the distinct textures at junctions are sufficiently different to enable the differentiation of various junctions. However, the specific texture chosen in our simulated environment may not fully capture the variations found in real-world pipeline networks. Factors such as corrosion, dirt accumulation, and wear over time can introduce significant variability in visual appearance, potentially affecting the performance of our NCC-based matching approach. Future work should therefore evaluate our method across a diverse range of textures to better understand its generalization capabilities and robustness in real-world applications. In summary, while our work emphasizes image-matching algorithms for active vision, it is crucial to recognize and address the challenges posed by the physical robot's limitations, system integration in real-world pipe networks, failure recovery, lighting conditions, and the need to validate the method with different pipe textures in physical pipe environments.

Chapter 4

Localization and Metrically Enhanced Topological Mapping via Panoramic Image Stitching

4.1 Introduction

In this chapter, we introduce an approach to enhance robotic localization and mapping within buried pipe networks by utilizing panoramic image stitching. This method aims to improve the accuracy of topological maps, which are critical for robotic navigation and decision-making processes in such intricate environments. By integrating panoramic imaging, the robot can capture comprehensive visual data at junctions, allowing for a detailed understanding of the spatial relationships and geometrical configurations of the network.

The concept of Metrically Enhanced Topological Mapping (METM) is introduced, which leverages the benefits of both topological and metric maps. Conventional topological maps offer an abstract representation of the environment by illustrating connectivity between significant points, such as junctions, through nodes and edges. However, these maps often lack precise geometric details, which are crucial for accurate localization and navigation in environments where small positional errors can lead to significant operational challenges.

Our approach involves stitching together panoramic images at critical junctions within the pipe network to derive precise angular relationships between different paths. This enhanced mapping process provides a more accurate representation of the physical layout of the network, enabling the robot to make more informed navigation decisions. By incorporating real-world geometric relationships into the topological framework, we improve the robot's ability to navigate pipe configurations, reduce localization errors, and optimize path planning.

This chapter details the development of the panoramic image stitching technique, the challenges associated with its implementation, and the integration of this method into the robotic system. We also present extensive experimental evaluations, including simulations and real-world trials, to validate the effectiveness and accuracy of the proposed method. The results demonstrate that the integration of panoramic image stitching significantly enhances the robot's mapping capabilities, leading to improved performance in navigation and task execution within buried pipe networks.

Metrically Enhanced Topological Mapping (METM)

In this work, we propose the term Metrically Enhanced Topological Mapping (METM) to describe a technique that enhances conventional topological maps by incorporating precise angular relationships between nodes. This method is designed to offer a more precise and informative representation of the environment, which can be particularly valuable for improving localization and navigation in intricate, confined spaces such as buried pipe networks. This approach maintains the high-level abstraction of topological maps, which represent environments through nodes (significant locations) and edges (connections between these locations), while embedding real-world angular information. By integrating true angular relationships, METM improves the system's ability to accurately navigate and localize within an environment, facilitating more effective path planning and spatial orientation. This method is particularly beneficial in pipe environments, enhancing the capabilities of autonomous systems such as mobile robots and inspection robots by providing both connectivity and accurate directional guidance.

4.2 Assumptions, Objectives and Contributions

This section outlines the assumptions, objectives, and contributions of the research in developing a real-time robotic localization and METM system for buried pipe networks using panoramic image stitching. It extends the assumptions, objectives, and contributions presented in Section 1.3 by incorporating additional objectives and contributions specific to the methodologies employed in this chapter.

Objectives

1. Stitch together images captured at each junction to create a panoramic view and compare this panoramic image with those from previous junctions to determine the robot's current location.

2. Develop a method for real-time robotic localization and generate metrically enhanced topological maps of buried pipe networks through panoramic image stitching.

3. Document the robot's orientation each time an image is captured to ensure the map includes precise angular relationships between nodes.

Contributions

1. Introduction of an approach that utilizes panoramic image stitching to compare unique locations within the pipe network, leveraging the wide field of view and resolution of panoramic images to accurately identify and differentiate junctions and other critical points.

2. Development of a METM system designed for pipe networks, representing network connectivity and including angular relationships between nodes for a metrically enhanced representation.

4.3 Problem Description

The technical problem addressed in this research involves the real-time localization and METM of autonomous robots operating in buried pipe networks, where conditions such as confined spaces and limited distinguishing features present significant challenges to conventional methods. The system focuses on identifying the robot's position at junctions using a combination of visual and distance sensor data to generate metrically enhanced topological maps that accurately represent the connectivity and angular relationships between junction exits.

The input data consists of high-resolution images (1920x1080 pixels) captured at regular intervals, alongside precise range measurements (with a ±1 cm tolerance). Odometry is utilized specifically to determine the robot's orientation upon reaching junctions, without attempting to address odometry drift. Due to the lack of distinctive visual features in the homogeneous pipe environment, correcting for drift is impractical. Instead, the system uses orientation data to assist with positioning at junctions.

The system's output is a topological map where nodes represent junctions and edges represent pipe segments. Angular information between junction exits is captured to ensure an accurate representation of the network's physical layout. Junction geometry and exit angles are determined through panoramic image stitching, which provides a comprehensive view of each junction and the angular relationships between exits.

Unlike methods that rely on template matching, this approach utilizes panoramic image stitching to extract the geometric features of the junction and the angles of each exit. The identification of the robot's current position is achieved by comparing the geometric relationships between the current junction and its exits with the spatial relationships recorded at the previous node, allowing the system to determine adjacency and connectivity within the network.

The problem is further constrained by the requirement for real-time operation, necessitating efficient data processing to ensure low-latency decision-making. The system updates the robot's pose and the topological map dynamically as the robot navigates the network, maintaining an accurate and up-to-date representation of both the network's structure and orientation.

4.4 Autonomous Control

The autonomous control system used in this research is based on the control algorithm developed by Nguyen et al. (Nguyen et al., 2022). As discussed in Section 3.2.1, this algorithm enables the robot to navigate buried pipe networks autonomously by utilizing a state-based decision-making process. For further details on the algorithm's implementation, please refer to Section 3.2.1, where it is thoroughly explained.

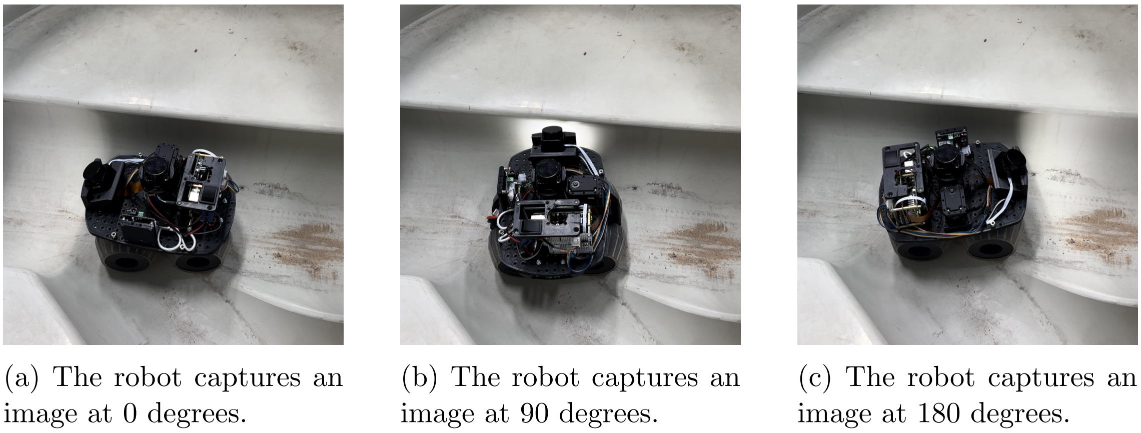
4.5 Panoramic Image Stitching

The development of a reliable and accurate panoramic image stitching method is an effective approach for improving the localization and mapping of robots within buried pipe networks. This section details the principles and methodologies of panoramic image stitching, focusing on capturing, processing, and combining images into panoramic views using the normalized cross-correlation (NCC) algorithm to determine the best stitching positions.

The process begins with the selective activation of the robot's vision sensors at critical points within the pipe network, such as junctions. The use of distance sensors helps identify these critical points, at which the vision sensors are activated to capture a series of images from different angles. This selective activation is designed to conserve energy and computational resources, as continuous image capture is unnecessary and inefficient in the confined and repetitive environment of pipe networks.

Once the robot arrives at a junction, it uses its IMU (Inertial Measurement Unit) to record the current orientation of the camera relative to its initial position. The IMU measures the robot's orientation through its gyroscope, which detects rotational movement around the three axes. These measurements are relative, meaning they provide the robot's orientation in reference to its starting point, rather than an absolute orientation. The IMU tracks changes in angular velocity over time to calculate the robot's orientation, with its accuracy depending on the quality of the sensor and the accumulation of drift over time. To mitigate errors, the robot can periodically recalibrate its orientation based on known reference points. This IMU is integrated into our custom-developed robot, Skatebot, as shown in Figure 4.9a. This initial angle is saved and will be used later to calculate the actual angle corresponding to each exit of the junction. After capturing this orientation data, the robot begins rotating its vision sensors to cover a 360-degree field of view. The rotation is carefully controlled to ensure sufficient overlap between consecutive images, which is essential for accurate stitching. Typically, the robot captures images at fixed angular intervals to ensure complete coverage of the junction. As illustrated in Figures 4.1a, 4.1b, and 4.1c, the robot captures images at angles of 0 degrees, 90 degrees, and 180 degrees, respectively. These figures serve as examples to illustrate the process. In practice, the robot captures images at 30-degree intervals, ensuring comprehensive coverage of the junction. By combining the initial IMU-recorded angle with the subsequent image captures, the robot can accurately map and determine the angles corresponding to each junction exit, facilitating precise mapping and navigation across the complete 360-degree view, as shown in Figure 4.2, where different angles of the junction are captured by the rotating camera on the robot.

The captured images are then processed to correct any distortions caused by the camera lens and the confined space of the pipe. This preprocessing step is vital to ensure that the images can be accurately aligned and combined. Common preprocessing techniques include lens distortion correction, image scaling, and

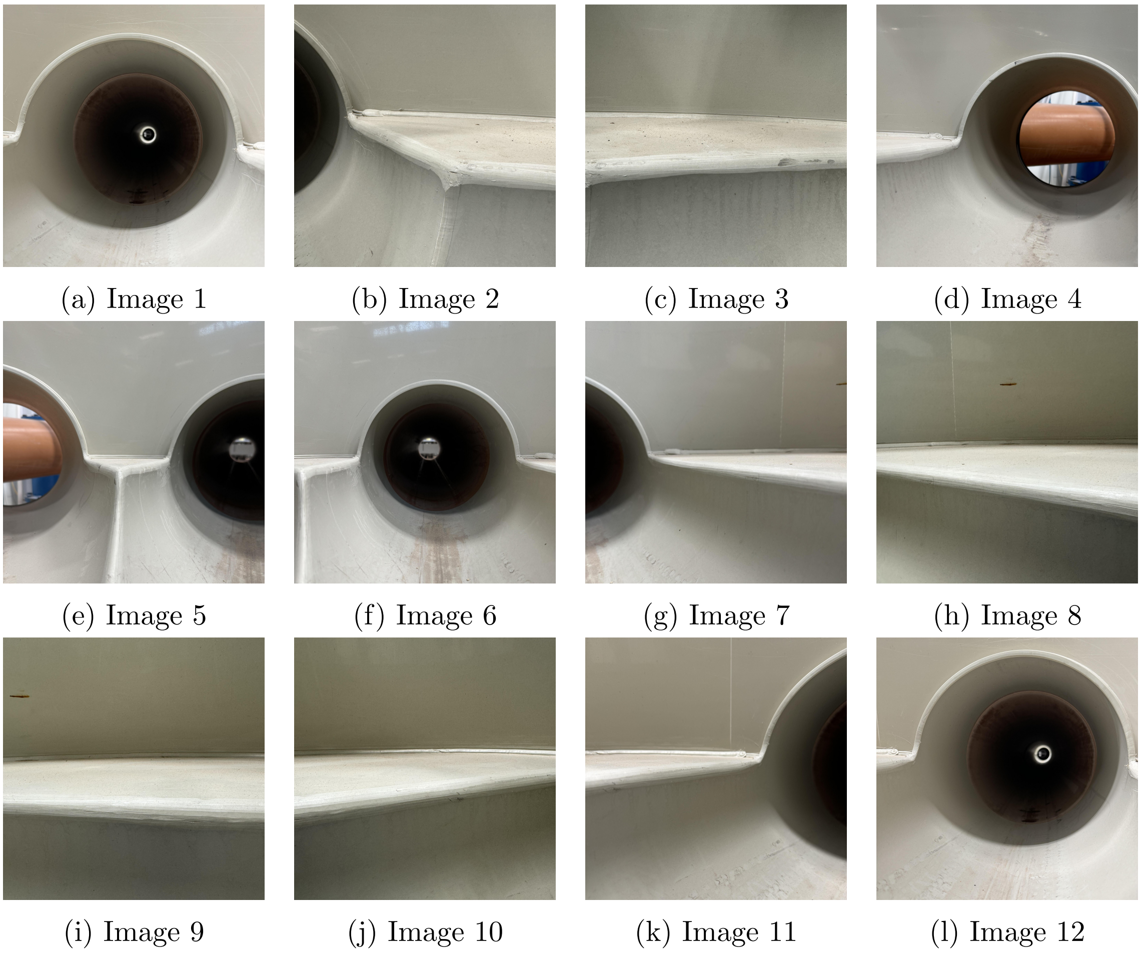


(a) The robot captures an image at 0 degrees.

(b) The robot captures an image at 90 degrees.

(c) The robot captures an image at 180 degrees.

Figure 4.1: The robot captures images by rotating its skate to cover a junction, showing different angles of vision capture.



(a) Image 1

(b) Image 2

(c) Image 3

(d) Image 4

(e) Image 5

(f) Image 6

(g) Image 7

(h) Image 8

(i) Image 9

(j) Image 10

(k) Image 11

(l) Image 12

Figure 4.2: Different angles of a junction captured by a rotating camera on a robot.

contrast adjustment to handle varying light conditions within the pipe.

The core of the panoramic image stitching process involves aligning the preprocessed images to create a seamless panoramic view. This is achieved through Algorithm 5:

Algorithm 5 creates a seamless panoramic image from a set of images captured at a junction within a pipe network. It starts by initializing an empty panoramic image P, which is progressively built by aligning and adding each image from the input set I. For each image, the algorithm computes the horizontal shift required for alignment with P using the NCC method. First, both the current image and the panoramic image P are converted to greyscale to simplify the comparison. The algorithm then identifies the rightmost Region of Interest (ROI) in the panoramic image where the new image is expected to align. The ROI is extracted from the panoramic image starting from the position where the new image width fits entirely within the rightmost part of the panoramic image. The NCC score is computed between the greyscale version of the current image and the ROI, generating an NCC matrix that measures similarity at various shift positions. The peak in the NCC matrix identifies the best matching position, and the corresponding shift is calculated. This initial shift is then adjusted to account for the starting position of the ROI within the panoramic image, ensuring accurate horizontal alignment. The adjusted shift $shiftX$ is applied to the current image, which is then added to the panoramic image, expanding the view. This process is repeated for each image in the input set I, and once all images are processed, the algorithm returns the final panoramic image. This approach ensures accurate alignment, creating a seamless and reliable panoramic image, which is crucial for effective localization and navigation in pipe networks.

The resulting panoramic image provides a detailed view of the junction, which is crucial for the robot's localization and mapping tasks. By comparing the panoramic images captured at different junctions, the robot can accurately determine its current location within the pipe network. This comparison involves matching the current panoramic image with those stored in the robot's database, using the NCC algorithm to ensure precise localization.

One of the key advantages of using panoramic images is their ability to capture a wide field of view with high resolution, providing detailed spatial information about the junction. This is particularly important in the environment of pipe networks, where precise localization is challenging due to the lack of distinctive landmarks and the repetitive nature of the surroundings.

The effectiveness of the panoramic image stitching method is validated through both simulations and real-world experiments. Simulations allow for the testing and refinement of the stitching algorithms in a controlled environment, while real-world deployments demonstrate the method's reliability and practicality in actual pipe networks. These experiments have shown that the panoramic image stitching method, using NCC for alignment, provides accurate and reliable localization, significantly enhancing the robot's ability to navigate and map the network.

We present the detailed results of these experiments in Section 4.8. For example, Figure 4.6 illustrates the panoramic image generated in a simulated pipe environment, demonstrating the method's ability to accurately capture the structure

Algorithm 5: Panoramic Image Stitching Algorithm

```
Panoramic Image Stitching ($I$);
Input  : Set of images $I = \{I_1, I_2, \ldots, I_n\}$ from the current junction
Output: Panoramic Image $P$
/* Initialize an empty panoramic image $P$ */
$P \leftarrow$ InitializePanoramicImage();
/* For each image, compute the horizontal shift using NCC and
   align it with the panoramic image $P$ */
foreach $image_i \in I$ do
    /* Convert both the current image and the panoramic image to
       greyscale */
    $greyImage \leftarrow$ ConvertToGreyscale($image_i$);
    $greyP \leftarrow$ ConvertToGreyscale($P$);
    /* Determine the region of interest (ROI) in $P$ where $image_i$
       is expected to align */
    $imageWidth \leftarrow$ Width($greyImage$);
    $panoramicWidth \leftarrow$ Width($greyP$);
    $ROI \leftarrow$ ExtractROI($greyP$, panoramicWidth - imageWidth,
     panoramicWidth);
    /* Compute the NCC matrix between the current image and the
       ROI in $P$ */
    $nccScore \leftarrow$ NCC($greyImage$, $ROI$);
    /* Find the peak in the NCC matrix, which gives the best
       match */
    $maxNCC \leftarrow nccScore$;
    /* Initially, we set maxNCC as the result from the first NCC
       computation */
    $shiftX \leftarrow 0$;
    /* Initial horizontal shift for the first alignment */
    /* For each possible horizontal shift (x-axis), compute the
       overlap between the current image and the ROI */
    for $x = 1$ to $Width(ROI) - 1$ do
        /* Slide the current image over the ROI and compute the
           NCC for each shifted position */
        $shiftedROI \leftarrow$ ShiftROI($ROI$, x);
        $shiftedImage \leftarrow$ ShiftImage($greyImage$, x);
        $nccScore \leftarrow$ NCC($shiftedImage$, $shiftedROI$);
        if $nccScore > maxNCC$ then
            $maxNCC \leftarrow nccScore$;
            $shiftX \leftarrow x$;
        end
    end
    /* Adjust the shift to account for the starting position of
       the ROI in $P$ */
    $shiftX \leftarrow shiftX + (panoramicWidth - imageWidth)$;
    /* Add the shifted image to the panorama */
    $P \leftarrow$ AddShiftedToPanorama($image_i$, $shiftX$, $P$);
end
return $P$;
```

and exits of junctions. Furthermore, Figure 4.10 provides an example from real-world experiments, showcasing a stitched panoramic image from the ICAIR Lab. These figures demonstrate how the panoramic image stitching method effectively supports METM and localization in both simulated and real-world environments.

4.6 Spatial Relationships

In this section, we discuss the methodology for determining spatial relationships within the pipe network, focusing on the logical process used to identify and differentiate nodes (junctions). The approach relies heavily on panoramic image analysis and angular measurements to build an accurate topological map of the network.

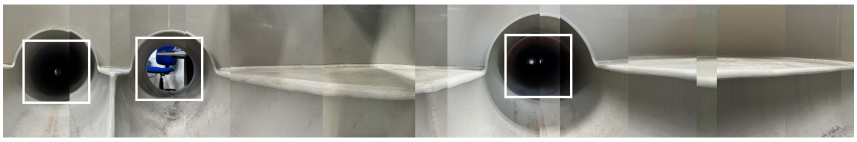


Figure 4.3: Template matching using NCC to identify exits.

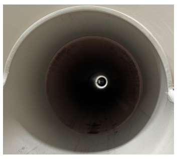


Figure 4.4: Template image used for matching.

In the panoramic image, the robot employs template matching techniques to identify the exit positions of the junction as shown in Figure 4.3. By performing NCC with a predefined exit template, peaks in the NCC matrix indicate potential exit locations. For each detected exit, the relative angle θ_{relative} is calculated using the formula:

$$\theta_{\text{relative}} = \left(\frac{x}{\text{width of panoramic image}} \right) \times 360 \text{ degrees} \tag{4.1}$$

where x is the x-coordinate of the peak in the NCC matrix.

To determine the actual angle θ_{actual} of each exit, the robot adds the initial angle θ_{imu} obtained from the IMU when the camera starts capturing images. The final exit angle is calculated as:

$$\theta_{\text{actual}} = \theta_{\text{relative}} + \theta_{\text{imu}} \tag{4.2}$$

As the robot moves through the pipe network, it constructs a topological map consisting of nodes (junctions) and edges (paths between them). Each node records the number of exits and their angular directions. To determine whether a junction is new or previously visited, the robot first compares the current junction's exit configuration (number of exits and their angles) with the stored configurations of all adjacent nodes from the previous node. If both the number of exits and the angles match within a defined tolerance, the robot further verifies the relative position

Algorithm 6: Node Identification and Differentiation

```
NodeIdentificationAndDifferentiation (Adjacent node list $A$);
Input  : Adjacent node list $A$
Output: Identification of junction as new or previously known
/* Step 1:  Image Capture and Exit Extraction                    */
Capture panoramic image $P$ at the current junction;
Call DetectExits($P, \theta_{\text{imu}}$) to extract exits $E_P = \{e_{P_1}, e_{P_2}, \ldots, e_{P_m}\}$ and their
  angular directions $\theta_P = \{\theta_{P_1}, \theta_{P_2}, \ldots, \theta_{P_m}\}$;
/* Step 2:  Comparison with Adjacent Nodes Using Exit Count and
   Angle Matching                                                */
foreach adjacent node $A_i \in A$ do
    Extract exits $E_{A_i} = \{e_{A_i1}, e_{A_i2}, \ldots, e_{A_in}\}$ and their angular directions
      $\theta_{A_i} = \{\theta_{A_i1}, \theta_{A_i2}, \ldots, \theta_{A_in}\}$;
    /* Step 3:  Exit Count and Angle Matching                    */
    if $|E_P| = |E_{A_i}|$ and Angles $\theta_P^{actual}$ and $\theta_{A_i}$ match within a defined
      tolerance then
        /* Step 4:  IMU-Based Relative Position Matching         */
        Compare the IMU angle from the current node to the previous node,
          $\text{IMU}_{\text{currentNode}}$, with the IMU angle from the candidate adjacent
          node $A_i$ to the previous node, $\text{IMU}_{A_i \to \text{prevNode}}$:

            $$\Delta\text{IMU} = |\text{IMU}_{\text{currentNode}} - (\text{IMU}_{A_i \to \text{prevNode}} + 180^\circ)|$$

        if $\Delta IMU \leq IMU\ Threshold$ then
            Identify as previously known node;
            return previously known node;
        end
    end
end
/* Step 5:  Identification of New Node                           */
Identify as new node;
return new node;
```

between the nodes using IMU data. If the relative position matches, the junction is identified as previously visited; otherwise, it is recorded as a new junction.

Once the exits are identified, the robot determines which exit to take next based on its navigation strategy. This strategy typically involves selecting the right-most unexplored exit to ensure systematic coverage of the network. The robot records the current junction's exits and their angles.

When the robot reaches the next junction, it again captures a panoramic image and identifies the exits and their angles. To determine if this junction is a new node or one that has been previously visited, the robot compares the current junction's exit configuration with the recorded information of all known connections from the adjacent node.

The algorithm 6 is designed to determine whether a junction in a navigation or mapping system is a known node or a new node. It begins by capturing a panoramic image P at the current junction and uses an exit detection algorithm to extract the exits $E_P = e_{P_1}, e_{P_2}, \ldots, e_{P_m}$ and their corresponding angular directions $\theta_P = \theta_{P_1}, \theta_{P_2}, \ldots, \theta_{P_m}$. This information is compared to the stored data of exits and angles from adjacent nodes $A = A_1, A_2, \ldots, A_n$. In this context, "adjacent nodes" refer to all nodes that are directly connected to the previously visited node. These adjacent nodes are identified based on their direct connection to the previous node through the graph structure representing the topological map. When the robot arrives at a new junction, it compares the current junction's exits and angles with those of all adjacent nodes of the previous node, rather than comparing the current junction to all nodes in the entire network. For each adjacent node A_i, the algorithm extracts exits $E_{A_i} = e_{A_i 1}, e_{A_i 2}, \ldots, e_{A_i n}$ and their angular directions $\theta_{A_i} = \theta_{A_i 1}, \theta_{A_i 2}, \ldots, \theta_{A_i n}$. If the number of exits at the current junction $|E_P|$ matches the number of exits at an adjacent node $|E_{A_i}|$ and their angular directions match within a defined tolerance, the algorithm proceeds to verify the relative position using IMU data.

First, the algorithm computes the angular difference between the current node and a potential match from the database. The potential match refers to a node or junction that has already been identified as geometrically similar based on the previously computed features, such as the number of exits and their angular directions. This is achieved by comparing the IMU angle recorded during the robot's passage through the junction to the IMU data of the adjacent nodes of the previous node (denoted as $IMU_{\text{adjacentNode}\to\text{prevNode}}$). Specifically, the algorithm calculates the absolute difference between the current node's IMU angle and the stored IMU angle of the adjacent node, adjusted by 180° (since the orientation could be reversed when revisiting nodes). The comparison is expressed as:

$$\Delta IMU = |IMU_{\text{currentNode}} - (IMU_{\text{adjacentNode}\to\text{prevNode}} + 180^\circ)|$$

If this difference, ΔIMU, falls within a predefined threshold, it confirms that the junction is a previously known node. If the difference exceeds the threshold, it indicates that the junction is new and should be recorded as such. This IMU-based relative positioning allows the robot to maintain an accurate and up-to-date map by cross-referencing orientation data from previous visits to junctions.

This process enables the algorithm to accurately distinguish between known and new nodes, dynamically updating the map or navigation system based on both exit configurations and relative positioning.

The algorithm 7 identifies exits in 360-degree panoramic images through a series of steps. Initially, the algorithm preprocesses the image by converting it to greyscale and applying a Gaussian blur to reduce noise. Then, it employs NCC to compare the greyscale image with a pre-defined exit template, producing an NCC matrix that highlights areas of similarity. The algorithm detects local maxima in the NCC matrix above a certain threshold, indicating potential exit locations. For each detected peak, it calculates the corresponding angle using the Formula 4.1 which maps the horizontal position to an angle within the 360-degree view. Finally, the algorithm returns a list of these angles, representing the positions of exits in the panoramic image.

Algorithm 7: Exit Detection from Panoramic Images

```
DetectExits ($P, \theta_{\text{imu}}$);
Input  : Panoramic image $P$ (360 degrees), IMU initial angle $\theta_{\text{imu}}$
Output: List of detected exits with their angles
/* Step 1:  Preprocessing                                    */
Convert $P$ to greyscale;
Apply Gaussian Blur to reduce noise;
/* Step 2:  Template Matching Using NCC                      */
Load the exit template $Temp$;
Perform NCC between $P$ and $Temp$;
$nccMatrix \leftarrow NCC(P, Temp)$;
/* Step 3:  Detect Peaks in NCC Matrix                       */
Find local maxima in $nccMatrix$ above a threshold;
foreach peak $(x, y)$ in $nccMatrix$ do
    /* Compute template center and angle for each detected exit
       */
    Compute template center $x_{\text{center}} \leftarrow x + \frac{\text{width}(Temp)}{2}$;
    Compute angle $\theta \leftarrow \left(\frac{x_{\text{center}}}{\text{width}(P)} \times 360 + \theta_{\text{imu}}\right)$ mod 360 degrees;
    Append $\theta$ to exit list;
end
/* Step 4:  Return Detected Exits                            */
return list of detected exits with their angles;
```

This logical process ensures that the robot can accurately distinguish between new and previously visited junctions, even in the repetitive environment of a pipe network. By continuously updating the topological map with detailed spatial relationships, the robot can maintain an accurate representation of the network's structure.

The topological map itself consists of nodes representing junctions, with edges indicating the paths between them. Each node includes information about the number of exits and their angular directions, while the edges store the direction and distance to the connected nodes. This detailed mapping allows the robot to navigate efficiently, avoid redundant exploration, and ensure coverage of the network.

In summary, the spatial relationship methodology involves capturing panoramic images at junctions, analyzing these images to identify exits and their angles, and using logical comparisons to distinguish between new and previously visited nodes based on all known connections. This process allows the robot to build a precise and reliable topological map of the buried pipe network, facilitating effective navigation and inspection tasks.

4.7 Metrically Enhanced Topological Mapping

METM is a critical component in the robotic navigation and localization system for buried pipe networks. This section details the methodology and processes involved in creating a topological map that accurately represents the spatial structure and connectivity of the pipe network, facilitating efficient navigation and inspection tasks.

The topological map is constructed as the robot explores the pipe network, using data collected from distance sensors and panoramic images. The algorithm 8 constructs a topological map G using panoramic images. It starts by initializing an empty graph G. For each panoramic image P, it determines if the image represents a new or known junction by comparing it with panoramic images of adjacent nodes A from the last node P_{last} using a separate node identification algorithm. If the junction is identified as new, a new vertex is added to the graph. The topological map G is then updated with new vertices and edges based on the identified junctions. Finally, the list of adjacent junctions A is updated, and the map G is returned.

The algorithm 9, is designed to update a topological map by integrating angle information extracted from panoramic images. The algorithm takes as input the current vertex, a new vertex, the existing topological map, and a panoramic image of the last node. It begins by extracting exits and their respective angles from the panoramic image. It then selects the exit whose angle is closest to 180 degrees within the 0-180 degree range. If the new vertex is not already present in the topological map, it adds the vertex based on the chosen exit's angle. Subsequently, if an edge between the current and new vertex does not exist, it adds this edge to the map, embedding the angle information. The updated topological map is then returned, reflecting these additions and modifications. This process aids in the creation of a structured map that incorporates directional information, enhancing navigation and mapping accuracy. As shown in the accompanying Figure 4.5, the map gradually expands by integrating new vertices and edges based on the extracted angle data, resulting in a progressively detailed representation of the environment.

The resulting topological map consists of nodes representing junctions and edges indicating the paths between them. Each node contains detailed information about the number of exits and their angular directions. The edges store data on the direction traveled between nodes, ensuring the map accurately represents the spatial layout of the pipe network.

By continuously updating the topological map with new data, the robot maintains an accurate representation of the network. This enables efficient navigation, as the robot can use the map to plan optimal routes, avoid redundant

```
Algorithm 8: Algorithm for Topological Mapping with Panoramic Images
TopologicalMappingWithPanoramicImages ($P, A$);
Input  : panoramic images $P$, Panoramic images of all adjacent nodes
         from the last node $A = \{A_1, A_2, \ldots, A_n\}$, panoramic images of the
         last node $P_{last}$
Output: Topological Map $G$
/* Initialize the topological map $G$ as an empty graph */
$G \leftarrow$ InitializeTopologicalMap();
/* For panoramic image $P$, identify if it represents a known or
   unknown junction */
$V_{current} \leftarrow NULL$;
/* Compare with all adjacent nodes */
/* Use Algorithm 6 to determine if the current junction is new
   or known */
$is\_new \leftarrow$ NodeIdentificationAndDifferentiation($A = \{A_1, A_2, \ldots, A_n\}$);
if $is\_new$ then
    $V_{new} \leftarrow$ AddNewVertex($panoramic\_image$, $G$);
end
/* Update the topological map $G$ with new vertices and edges */
$G \leftarrow$ UpdateTopologicalMap($V_{current}$, $V_{new}$, $G$, $P_{last}$);
/* Update the adjacent junction list */
$A \leftarrow$ GetAdjacentJunctions($V_{new}$, $G$);
return $G$;
```

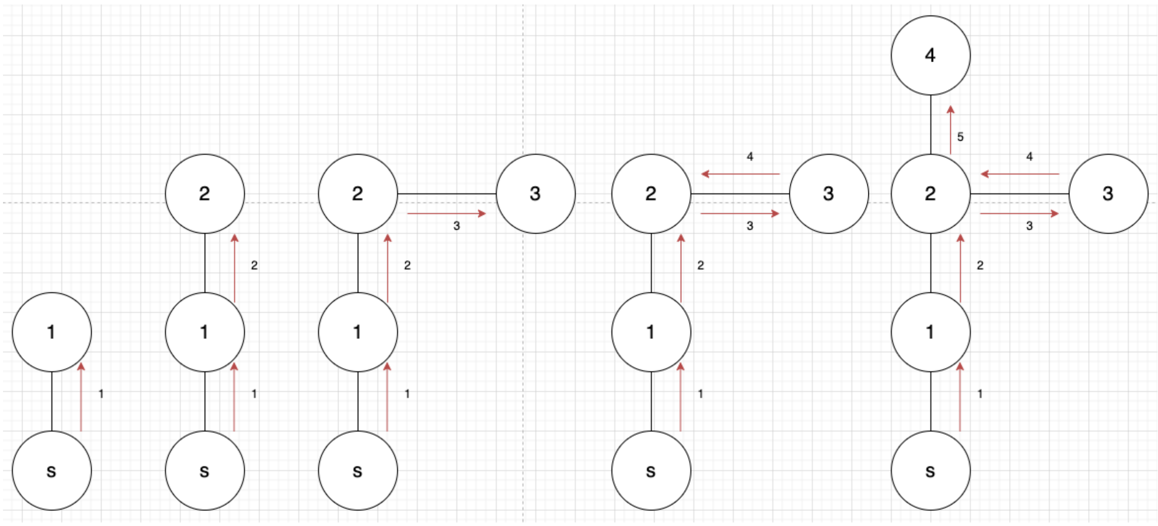


Figure 4.5: The flowchart illustrates the gradual expansion of a topological map, as new nodes and edges are added based on angular information from panoramic images.

Algorithm 9: Update Topological Map with Angle Information

```
UpdateTopologicalMap ($exit\_list, V_{current}, V_{new}, G$);
Input  : Extract exits from the panoramic image $exit\_list$, Current vertex
         $V_{current}$, New vertex $V_{new}$, Topological Map $G$
Output: Updated Topological Map $G$
/* Step 1:  Choose the exit with angle closest to 180 degrees
   within 0-180 range                                              */
$chosen\_exit \leftarrow NULL$;
$min\_angle\_diff \leftarrow 180$;
foreach $exit \in exit\_list$ do
    if $0 \leq exit.angle \leq 180$ then
        $angle\_diff \leftarrow |180 - exit.angle|$;
        if $angle\_diff < min\_angle\_diff$ then
            $min\_angle\_diff \leftarrow angle\_diff$;
            $chosen\_exit \leftarrow exit$;
        end
    end
end
/* Step 2:  Add vertex based on chosen exit's angle                */
if $V_{new}$ is not in $G$ then
    Add $V_{new}$ to $G$ at angle $chosen\_exit.angle$;
end
/* Step 3:  Add edge with chosen angle information                 */
if edge between $V_{current}$ and $V_{new}$ does not exist in $G$ then
    Add edge $E(V_{current}, V_{new}, chosen\_exit.angle)$ to $G$;
end
return $G$;
```

exploration, and ensure coverage of the network.

The topological mapping method has been validated through extensive simulations and real-world experiments. Simulations provide a controlled environment for testing and refining the algorithms, while real-world deployments demonstrate the system's reliability and practicality in actual pipe networks. These experiments have shown that the topological mapping method can accurately represent pipe networks, facilitating effective navigation and inspection.

In summary, topological mapping is a fundamental aspect of the robotic system, enabling precise and reliable navigation within buried pipe networks. By capturing and analyzing spatial relationships at junctions, recording angular directions, and systematically exploring the network, the robot constructs a detailed topological map that supports efficient and effective maintenance and inspection tasks.

4.8 Experiments & Results

4.8.1 Hypotheses

The study on real-time robotic localization and metrically enhanced topological mapping in buried pipe networks posits several key hypotheses that underpin the development and validation of the proposed methodologies:

1. The first hypothesis posits that the panoramic image stitching method can accurately identify and differentiate junctions within a pipe network. Accurate junction identification is fundamental for precise localization and navigation. The panoramic image stitching method, which leverages the normalized cross-correlation (NCC) algorithm, is expected to provide high-resolution, detailed images that capture the unique spatial configurations of each junction. By comparing these images, the robot should be able to distinguish between different junctions accurately, even in repetitive environments.

2. The second hypothesis is that the METM system can effectively represent the spatial layout of a pipe network, including both connectivity and precise angular relationships between nodes. This system is anticipated to reflect the actual geometric orientations within the network, providing a reliable framework for navigation. The hypothesis emphasizes the importance of geometric accuracy in the map, which is achieved by recording the robot's orientation each time an image is captured. This ensures that the angular relationships between nodes are well-documented and accurately represented in the topological map.

These hypotheses collectively aim to validate the proposed methodologies for enhancing the efficiency and reliability of robotic localization and mapping in buried pipe networks. By addressing these hypotheses through rigorous experimentation and analysis, the research seeks to demonstrate the practical viability and advantages of the developed systems. The outcomes of these tests will provide valuable insights into the effectiveness of the panoramic image stitching method and the accuracy of the METM system in real-world applications.

4.8.2 Experimental setup

The experimental setup was meticulously crafted to test the proposed hypotheses rigorously in both simulated and real-world environments. For the purpose of this study, a mobile robotic platform was developed and equipped with various sensors, including distance sensors, a high-resolution RGB vision system, and an inertial measurement unit (IMU). The robotic platform was tasked with autonomously localizing and mapping pipe networks.

The vision system, critical for image-based navigation, captured high-resolution images of pipe junctions, which were then processed using NCC techniques to build a panoramic image. The IMU assisted with orientation and movement tracking within the pipe networks, while the distance sensors were deployed to detect and measure the distance from the robot to the surrounding pipe walls, crucial for navigating through tight and confined spaces.

The experimental validation was performed in two distinct environments: simulations and real-world physical testing. Simulations, presented in Section 4.8.3, were executed using real image data and accurately modeled pipe environments that included variations in pipe network and junction configurations. The simulated environments closely mirrored real-world conditions, with the inclusion of typical variations found in buried pipe networks, such as T-junctions, Y-junctions, curves, and different environmental factors like lighting and debris. However, it is important to note that the number of junctions and the complexity of the pipe network configurations were constrained by the limitations of the experimental pipe conditions. Consequently, the simulations could not encompass more complex or irregular configurations that may be encountered in larger, real-world pipe networks.

In parallel to the simulations, real-world experiments, presented in Section 4.8.4, were conducted in a controlled environment set up at the ICAIR (The Integrated Civil and Infrastructure Research Centre) Laboratory. The testbed was designed to simulate an underground pipe network, with known dimensions and configurations representing typical challenges encountered in buried infrastructures. However, the testbed was intentionally simplified to include a limited number of junction types and obstacles, which may not fully represent the variety of real-world pipe network configurations encountered in larger and more complex systems. The physical testbed included a range of junction types and pre-placed obstacles to test the robot's ability.

The control algorithms and image processing techniques were implemented in Python, leveraging tools such as OpenCV and SciPy for efficient computation. The Robot Operating System (ROS) was employed to integrate and manage the robotic system's hardware and software components. ROS facilitated communication between the various sensors, motors, and control algorithms, enabling seamless operation of the robotic platform. The navigation algorithm was built around a finite state machine (FSM) to manage the robot's decision-making at junctions and to ensure efficient and exhaustive exploration of the pipe network. The NCC algorithm facilitated image matching, while the FSM handled autonomous navigation decisions, ensuring the robot could correctly interpret its surroundings and make appropriate decisions in real time.

The key performance metrics included the accuracy of the topological map generated by the robot and the reliability of junction identification. These metrics were evaluated throughout the robot's operation in both the simulated and real-world environments, providing insights into the system's robustness, accuracy, and potential for scalability to larger and more pipe networks. The results obtained from both simulation and real-world testing are expected to inform future improvements and adaptations of the system, ensuring its capability to handle more diverse and challenging pipe network configurations in practical applications.

4.8.3 Real Data-Driven Simulation

Validation of Panoramic Image Stitching

In this experiment, we evaluated the effectiveness of our panoramic image stitching method in accurately recognizing the number and angles of exits at each junction within the pipe network. The goal was to assess the precision of junction identification when panoramic images are used.

The experimental procedures involved collecting numerous high-resolution images from within actual pipes, and capturing various pipe junctions in real environments. These images, taken under controlled conditions with a high-resolution camera, depicted junctions with different numbers of exits and angles. We manually combined these real images to simulate 10 distinct pipe scenarios, each featuring 1 to 4 exits at different angles, mimicking real-world junction configurations. These images were then seamlessly stitched together using our Algorithm 5. The stitched panoramic images were analyzed using Algorithm 7. It is important to emphasize that all the images used in the simulations were sourced from actual robots operating in real pipe environments, and the simulations were generated through different combinations of these real images.

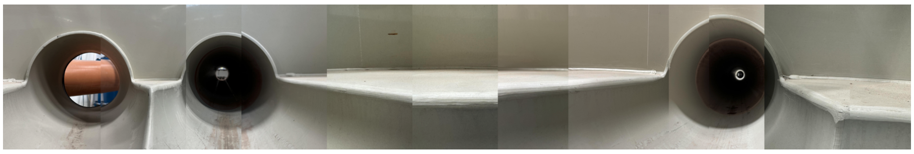


Figure 4.6: This is a panoramic image artificially generated by rearranging the sequence of collected images to form a pipe structure. The panoramic view shows a junction with three exits at different angles, allowing for accurate identification and measurement of the angles between the exits.

As shown in Figure 4.6, the panoramic image clearly illustrates a junction with three distinct exits, allowing for accurate identification and angular measurements of the exits in a pipe environment.

The panoramic image stitching experiment was conducted to validate the system's capability of accurately identifying exits and their angles at pipe junctions. The experiment was carried out in a simulated environment that closely replicated real-world conditions, utilizing real images of various pipe junctions. These junctions ranged from configurations with 1 to 4 exits, designed to simulate typical underground pipe scenarios.

The robot utilized a high-resolution camera to capture 36 overlapping images at each junction, rotating 10 degrees between each image to cover a full 360-degree view. This setup ensured significant overlap between images, which was necessary for accurate panoramic stitching. The captured images were processed using Algorithm 5.

The system was evaluated in fifteen different junction configurations. Each junction contained a varying number of exits and exit angles. These panoramic images were analyzed to detect the number of exits and measure the angles between them. The system's performance was assessed using two primary metrics: Exit Count Accuracy and Angular Accuracy.

The Exit Count Accuracy was measured by comparing the number of exits detected from the panoramic image with the actual number of exits present at the junction. Similarly, the Angular Accuracy was assessed by comparing the detected angles between the exits with the known angles. The panoramic image stitching method achieved a 100% accuracy in detecting the number of exits and a ± 9.5° accuracy in measuring the angles.

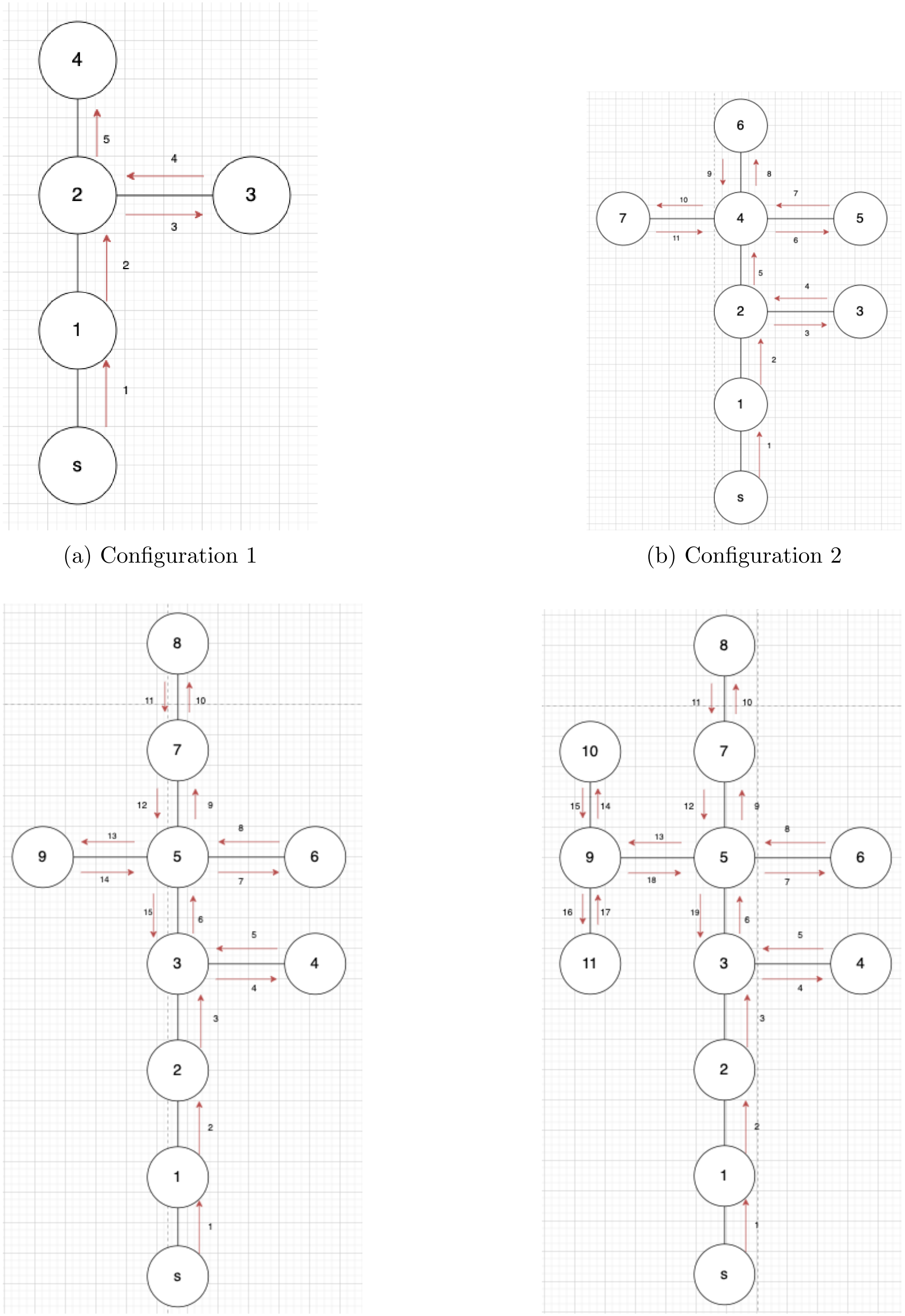
These experiments demonstrated that the panoramic image stitching method is highly reliable for exit detection and angular measurement. The setup effectively simulated real-world conditions, making it a possible method for applications in underground pipe networks.

Metrically Enhanced Topological Mapping Accuracy

In this section, we discuss the experimental design and results of our METM accuracy tests for underground pipe networks using a robotic system. We designed four distinct underground pipe network configurations to test our METM algorithm. Each configuration included multiple junctions with different shapes and layouts. The experimental setup involved a robot navigating through these junctions, capturing images at each junction, and updating the topological map based on the collected data. The four shapes tested are illustrated in the Figures 4.7. Each configuration represents a different complexity level in terms of junction shapes and connectivity.

At each junction, the robot utilized artificially generated high-resolution panoramic images. These images were then processed using our algorithm to identify the junction type and orientation. The algorithm's ability to accurately identify junctions and update the topological map was tested. This involved verifying the accuracy of junction identification and the precision of the recorded angular directions between nodes.

The generated topological maps of the four distinct topological network configurations, illustrated in Figure 4.8, provide a comprehensive view of the experimental setups used to validate the mapping algorithm. The experimental results demonstrate that the METM algorithm is effective in accurately identifying junctions and their orientations across various underground pipe network configurations. The results demonstrated that the algorithm consistently achieved completeness in the generated networks, effectively mapping the entire network structure, even as complexity increased. These findings indicate the reliability of the METM system, providing a foundation for its deployment in real-world applications.



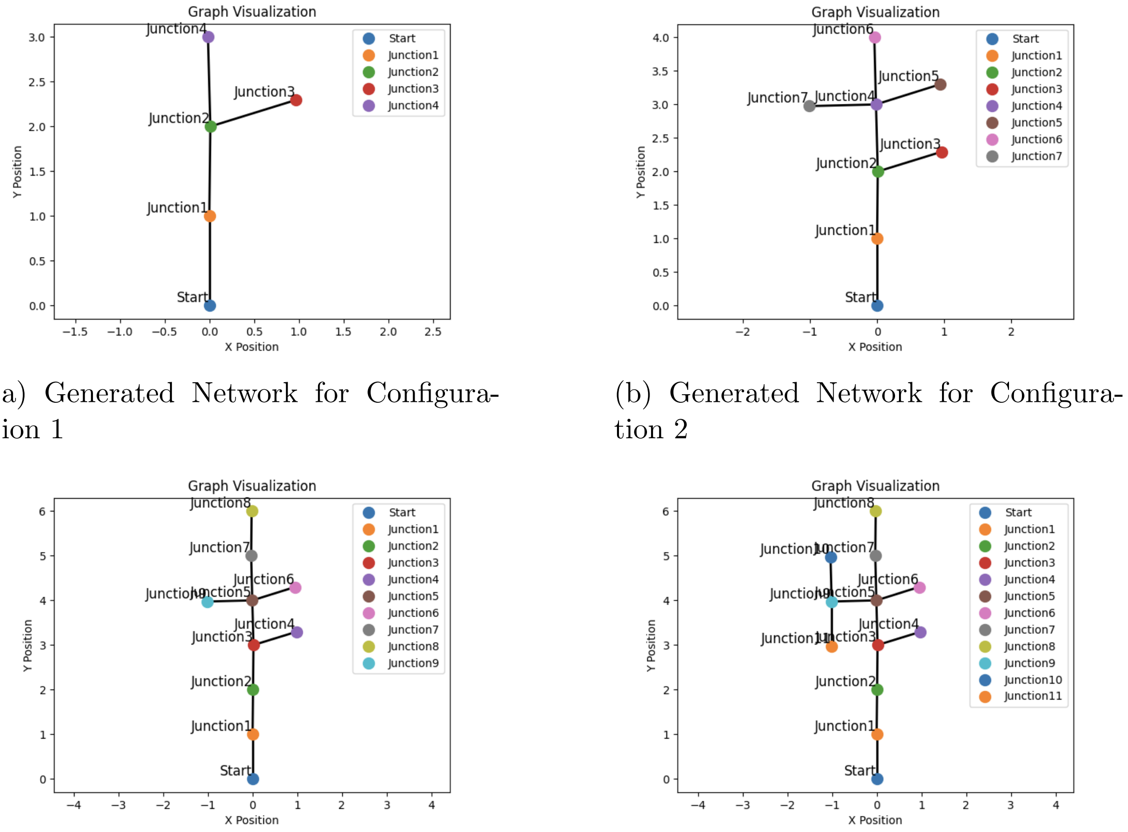
(a) Configuration 1

(b) Configuration 2

(c) Configuration 3

(d) Configuration 4

Figure 4.7: Different configurations of underground pipe networks.



(a) Generated Network for Configuration 1

(b) Generated Network for Configuration 2

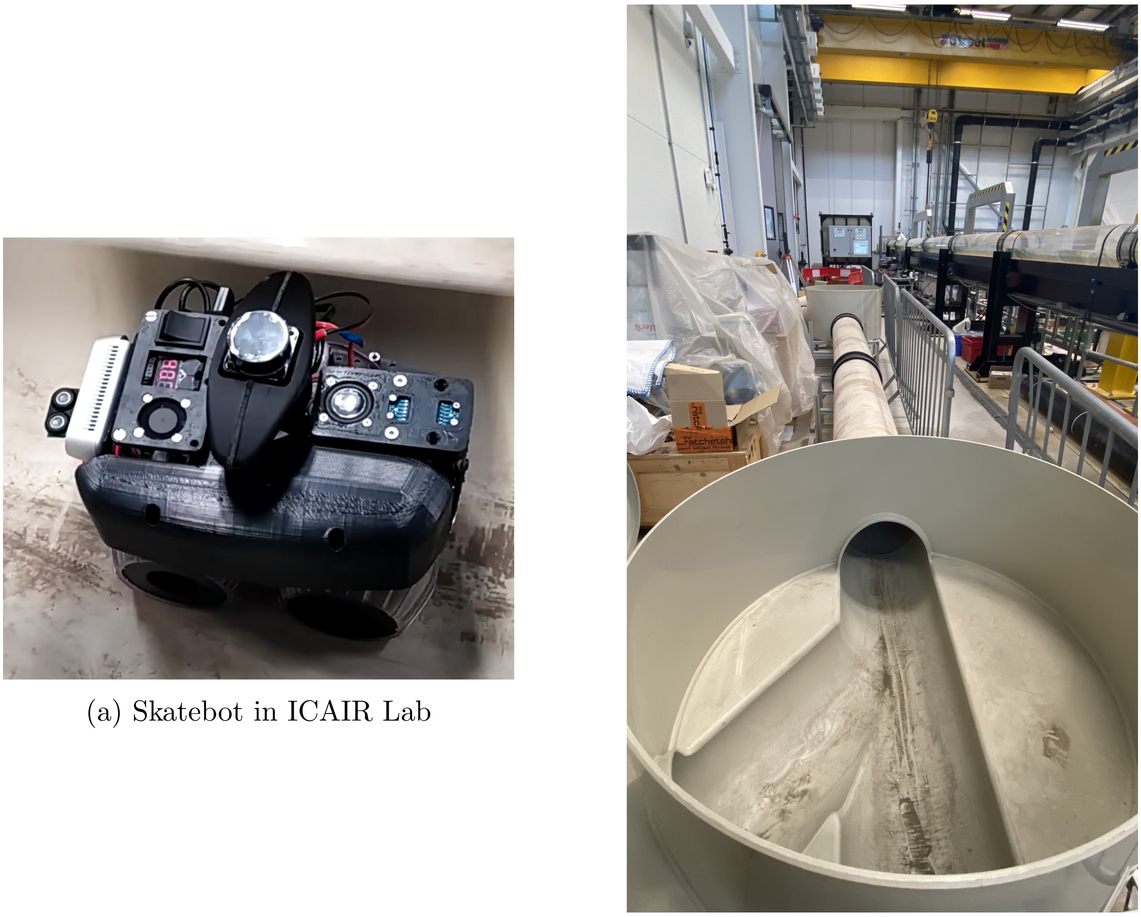
(c) Generated Network for Configuration 3

(d) Generated Network for Configuration 4

Figure 4.8: Generated topological networks for different configurations of underground pipe networks.

4.8.4 Real-robot results

In this section, we present the experimental results of testing the real robot, Skatebot (Figure 4.9a), in ICAIR's real pipe environment (Figure 4.9b). Skatebot is a collaborative development by the Pipebot project team, designed specifically for pipe inspection and mapping. The robot integrates many sensors, including LiDAR, cameras and acoustic sensors. It features front and rear LED lights for operation in dark environments, a rotatable platform for enhanced maneuverability, and a four-wheel drive system for navigating pipe networks. Powered by an onboard battery with an LED level display, Skatebot can function both autonomously and via manual remote control. Its core control system is built around a Raspberry Pi 4B, which manages its sensors and controls, enabling automated exploration and METM of pipe environments. The ICAIR pipe setup features only two junctions, but we conducted multiple tests from different entry points. Four different entry points were used for the tests. One junction was under illuminated conditions, while the other junction was in complete darkness. Each test iteration was conducted three times to ensure consistency.



(a) Skatebot in ICAIR Lab

(b) ICAIR Pipe Setup Overview

Figure 4.9: Skatebot in ICAIR Lab and ICAIR Pipe Setup

The real-world experiments were designed to rigorously test the hypotheses outlined in our study. The Skatebot was equipped with distance sensors, vision sensors, and an inertial measurement unit (IMU). The vision sensors captured high-resolution images, while the distance sensors detected junctions and measured distances to the pipe walls. The experimental environment was a controlled testbed replicating actual pipe network conditions with variations in junction types.

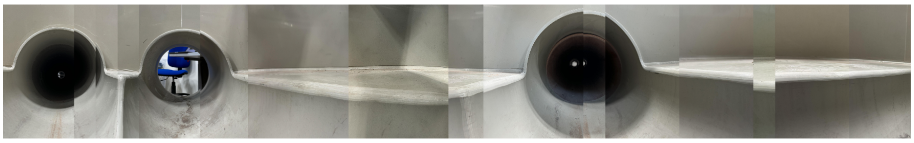


Figure 4.10: Example of a stitched panoramic image from the real-robot experiment. This panoramic view shows a junction with three exits at different angles.

Due to the current unreliability of the autonomous navigation algorithm on the Skatebot, we used a joystick to mimic the robot's autonomous exploration route in the pipe environment. The robot navigated through the ICAIR pipe setup, capturing high-resolution panoramic images, as shown in Figure 4.10, at each junction. These images were processed using our panoramic image stitching algorithm, which leveraged the normalized cross-correlation (NCC) method for precise image alignment. The algorithm identified junctions and measured angles between exits.

Each test iteration demonstrated the robot's capability to generate a metrically enhanced topological map of the pipe network. The generated maps, as seen in Figure 4.11, showed that the robot consistently identified the junctions and their orientations correctly. Even with the limited number of junctions, the robot's performance in creating reliable topological maps underscores its potential for deployment in more pipe networks.

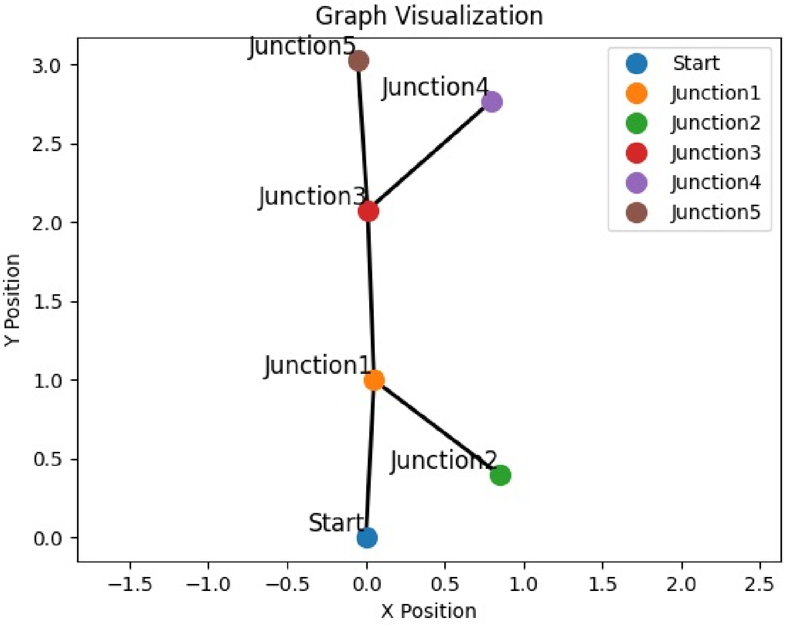


Figure 4.11: Example of a topological map generated by Skatebot in ICAIR Lab

The results from the real-robot experiments in ICAIR's pipe environment highlight the system's accuracy in METM. During these tests, the system consistently identified junctions correctly, and the angle measurements between junction exits remained within a tolerance of ±10 degrees. This was achieved across various tests, even when conditions varied, such as one junction being illuminated and another in complete darkness. These findings confirm the system's reliability and precision in constructing metrically enhanced topological maps of underground pipe networks.

4.8.5 Limitations

Despite the results obtained from both simulation and real-world experiments, the research has several limitations that must be acknowledged. These limitations highlight areas for future improvement and the challenges that need to be addressed for practical implementation in diverse and challenging environments.

Although the system demonstrated real-time processing capabilities in controlled environments, its performance under high computational loads needs further validation. In larger networks, the computational demands for image processing, data analysis, and decision-making may increase significantly. Ensuring that the system can maintain real-time performance without excessive delays or resource consumption is crucial for practical applications.

Energy consumption is another area of concern. While the selective activation of vision sensors conserved energy and computational resources, the overall energy efficiency of the system in long-duration missions remains a challenge. The robot's battery life and energy management strategies need to be optimized to support extended operations in extensive pipe networks. Future research should focus on developing more energy-efficient hardware and algorithms to prolong operational time.

Furthermore, the current system assumes relatively stable environments for accurate mapping. However, real-world pipe networks can change over time due to maintenance activities, natural wear and tear, and environmental factors. The system's ability to adapt to and update the topological map in response to these changes needs to be investigated further. Continuous and adaptive mapping strategies could enhance the system's long-term reliability and accuracy.

The practical implementation of the system also requires seamless integration with existing pipe network infrastructure, including compatibility with various pipe materials, sizes, and layouts. Developing standard protocols and interfaces for integration with different types of infrastructure will be essential for widespread adoption.

Lastly, in real-world scenarios, the presence of unexpected obstacles and debris can significantly impact the robot's localization. The current system's ability to detect and avoid such obstacles needs further refinement. Implementing obstacle detection and avoidance algorithms could improve the robot's reliability and safety during operations.

In summary, while the research presents a method for real-time robotic localization and metrically enhanced topological mapping in buried pipe networks, addressing these limitations will be crucial for deploying the system in diverse and challenging real-world environments. Future work should focus on enhancing environmental robustness, scalability, sensor reliability, real-time processing capabilities, energy efficiency, adaptive mapping strategies, and integration with existing infrastructure. By overcoming these challenges, the system can be further developed to provide reliable and efficient solutions for the maintenance and inspection of underground pipe networks.

4.9 Conclusion

This research has presented an approach to real-time robotic localization and metrically enhanced topological mapping within buried pipe networks using panoramic image stitching. The study focused on developing and validating methodologies that address the unique challenges posed by the confined environments of underground pipe networks.

The first major contribution of this work is the integration of panoramic image stitching for the identification and differentiation of junctions. By leveraging the normalized cross-correlation (NCC) algorithm, the system accurately captures high-resolution panoramic images, enabling precise identification of junctions and exits. This method demonstrated high success rates in simulations and effectively supported the robot's localization tasks.

The second contribution is the development of a METM system that not only represents the connectivity of the network but also includes precise angular relationships between nodes. This approach ensures that the map accurately reflects the physical layout and orientations within the pipe network, facilitating reliable navigation and decision-making.

Simulation results supported all two hypotheses, demonstrating that the system could accurately identify junctions, generate precise topological maps. The validation through both controlled simulations and real-world experiments underscores the practicality and reliability of the proposed methodologies. In real-world tests, the robot navigated the testbed environment, confirming the system's ability to handle actual pipe network conditions. The real-world experiments provided further evidence of the system's robustness and adaptability, although they also highlighted areas needing improvement.

Despite the outcomes, the research identified several limitations that must be addressed in future work. These include the need for enhanced environmental robustness, scalability, sensor reliability, real-time processing capabilities, and energy efficiency. Additionally, the system's ability to adapt to changing environments and integrate seamlessly with existing infrastructure requires further investigation.

In summary, this research provides a framework for real-time robotic localization and metrically enhanced topological mapping in buried pipe networks. By addressing the identified limitations and continuing to refine the system, it holds potential to enhance the efficiency and reliability of maintenance and inspection operations in underground infrastructure. The findings provide insights and techniques to the field of robotics and autonomous systems, facilitating future advancements in navigating and managing confined environments.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this research, we introduced and validated two methods for real-time robotic localization and metrically enhanced topological mapping in buried pipe networks. The key challenge was addressing the limited mobility and power constraints of miniature robots in confined environments. Our work leveraged multimodal sensing strategies, combining distance sensors and vision sensors, to enhance the robot's navigation and mapping capabilities.

The first method focused on localization through efficient junction detection and mapping. By using distance sensors to identify junctions and activating vision sensors only at critical points, we reduced computational load and minimized reliance on continuous video processing. This approach employed a localization algorithm utilizing convolutional filters for image matching, enabling the robot to identify and differentiate junctions by comparing newly captured images with a pre-existing database. This method demonstrated the robot's ability to explore and map the pipe network, reducing cumulative localization errors and improving overall navigation accuracy.

Building on the first method, the second method enhanced mapping accuracy by creating panoramic images from junction photos. These panoramic images provided a view of the junctions, allowing for precise identification of the number and angles of junction exits. By comparing the robot's trajectory and passed junctions, this method generated a topological map that included real-world geographic relationships, offering a more accurate representation of the physical layout of the pipe network. This refined mapping capability facilitated improved navigation and planning, and was validated through extensive simulations, showcasing its effectiveness in real-time localization of pipe networks.

The results from implementing and testing these methods in controlled simulations and small-scale real-world experiments demonstrated the feasibility of the proposed approaches. However, these findings were based on relatively idealized conditions, which may not fully capture the variability and complexity encountered in larger, more unpredictable real-world pipe networks. While the system effectively identified junctions and generated accurate topological maps, the real-world tests revealed areas for improvement, particularly under more challenging environmental conditions. Although these tests confirmed the system's adaptability to certain pipe network conditions, the limited range of the tests and the controlled experimental setup may not fully replicate the complexities of larger networks. Factors such as environmental fluctuations, pipe corrosion, and debris presence were not extensively tested, which underscores the need for further exploration to evaluate the system's performance in more diverse and challenging environments.

5.2 Contributions

This research makes several key contributions to the field of robotic localization and metrically enhanced topological mapping in buried pipe networks. A localization method has been developed, integrating distance sensing with selective visual sampling at junctions. This method optimizes the localization process by reducing the need for continuous visual input while maintaining sufficient accuracy. The application of Normalized Cross-Correlation (NCC) for junction recognition enables the system to distinguish between known and unknown junctions in real time, contributing to the overall reliability of localization within the network.

Additionally, the research proposes a method for generating metrically enhanced topological maps that represent both the connectivity and spatial relationships between junctions. By using panoramic image stitching at junctions, the robot can estimate the number of exits and their angles, providing a more detailed and accurate layout of the pipe network. This method improves navigation and supports decision-making in tasks like inspection and maintenance.

The system's performance was validated through simulations and physical experiments in small-scale pipe networks, demonstrating its ability to localize junctions and generate topological maps in real time. These results provide a foundation for future work in larger and more challenging pipe networks.

5.3 Limitations

While this research presents promising results for real-time localization and metrically enhanced topological mapping in buried pipe networks, several limitations must be acknowledged. The controlled environments used in simulations and physical tests, particularly with small-scale pipe networks, do not fully capture the variability and complexity of real-world pipe networks. These experimental setups were relatively idealized and less challenging compared to the extreme and unpredictable conditions typically encountered in larger, more complex networks. In real-world environments, factors such as temperature fluctuations, high humidity, corrosive substances, and varying levels of debris can introduce significant uncertainties, which were not extensively tested in the current setup. As such, the system's performance in more challenging and unpredictable conditions remains uncertain.

Additionally, the scalability of the system to larger, more intricate pipe networks with numerous junctions and varying pipe diameters has not been fully evaluated. While the system performed effectively in the controlled, small-scale networks used in these experiments, the performance in more complex and larger-scale networks, where computational load, energy consumption, and system reliability could be stressed, has not been thoroughly explored. Furthermore, conditions such as sediment build-up, strong external vibrations, or other real-world interferences were not extensively tested in this study.

Another limitation lies in the system's reliance on selective visual sampling and distance sensors. In environments with obstructed paths or low visibility, such as those with dirt, water, or physical barriers, these sensors may underperform, potentially leading to localization errors. The absence of multi-sensor fusion—such as integrating acoustic, thermal, or radar sensors—further limits the system's robustness in such challenging environments. While future work may address this by incorporating additional sensor modalities, the current system remains vulnerable to sensor degradation or environmental obstructions.

Furthermore, the control algorithms were optimized for specific, idealized conditions and may not generalize well to more diverse and unpredictable real-world pipe networks without further refinement. Additionally, the lack of optimization for real-time computational resources raises concerns about the system's scalability for larger networks or long-term operations, particularly in energy-constrained scenarios requiring extended deployments.

Finally, while the topological maps generated by the system are effective for mapping static environments, they do not yet account for dynamic changes, such as ongoing maintenance or the addition of new sections to the network. This limits the system's adaptability and long-term reliability in real-world applications where environmental changes are frequent.

In conclusion, although the proposed system offers a viable solution for real-time localization and mapping in buried pipe networks, addressing these limitations is crucial to ensure its effectiveness in more challenging, larger-scale environments. Future research should focus on testing the system in larger, more complex networks, incorporating multi-sensor fusion, and refining control algorithms to handle diverse, real-world conditions.

5.4 Future Work

Future developments could focus on several key areas to address the limitations identified in this study while building on the successes of the current research. First, integrating distance metrics into topological maps would provide a more detailed and accurate representation of the network layout. Incorporating a metric dimension for each node could improve both localization and mapping accuracy, particularly in more complex pipe configurations. This addition would allow for more precise distance measurements and better support the navigation of robots in challenging environments.

Another practical improvement would be to enhance the stability of control algorithms, particularly in response to environmental changes or network irregularities. Currently, the system has been shown to perform well under controlled conditions, but the ability to handle dynamic conditions, such as sudden changes in pipe orientation or network topology, remains an area for improvement. Future work should focus on refining the control algorithms to ensure consistent and reliable performance, even when the network undergoes unexpected shifts.

A crucial next step would be testing the system in larger, more complex pipe networks. While the current research demonstrates the system's effectiveness in small-scale experiments, the scalability of the system in more extensive environments, with varying pipe diameters and greater numbers of junctions, must be thoroughly assessed. This would help determine the system's efficiency and reliability in real-world scenarios, where network complexity is much higher.

In addition, developing adaptive mapping techniques that can update topological maps in real time as the network changes over time will be an important step forward. This adaptability would allow the system to handle ongoing changes due to maintenance, environmental degradation, or the addition of new sections to the network, ensuring that the maps remain relevant and accurate over time.

Lastly, integrating additional sensor modalities, such as acoustic sensors or radar, could significantly improve the robustness of the system, especially in environments where visibility is low or physical barriers obstruct sensors. Such multi-sensor fusion would enhance the system's ability to operate in diverse and challenging conditions, making it more adaptable to the unpredictable nature of real-world pipe networks.

In summary, while the current research has demonstrated the feasibility of real-time localization and topological mapping in controlled settings, future work should focus on addressing the system's scalability, adaptability, and robustness in larger, more dynamic environments. By incorporating distance metrics, refining control algorithms, testing in larger networks, and integrating additional sensors, the system could evolve into a more reliable and effective tool for maintaining and inspecting complex underground pipe networks.

5.5 Final Thoughts

The development of robotic systems capable of localization and mapping confined environments such as underground pipe networks has the potential to revolutionize infrastructure maintenance. The methodologies developed in this research, especially those centered on selective visual sampling and metrically enhanced topological mapping, pave the way for more efficient and reliable robotic solutions. As we move forward, further advancements could focus on the integration of multi-modal sensor fusion, including acoustic, thermal, or radar sensing, to increase robustness in visually challenging environments. Future iterations of this system may also benefit from adaptive mapping strategies that dynamically adjust to ongoing changes in pipe networks, such as infrastructure modifications or environmental factors. Moreover, exploring real-time optimization of computational resources and energy efficiency will be key in extending the operational capabilities of autonomous systems. These enhancements could impact not only the field of infrastructure robotics but also broader applications in other confined and hazardous environments like underground mines or industrial systems. Ultimately, the work presented in this thesis offers valuable insights and techniques that can be further developed to enhance the capabilities of autonomous robotic systems, contributing to their evolution into more resilient and intelligent solutions.

Bibliography

Aitken, J. M., Evans, M. H., Worley, R., Edwards, S., Zhang, R., Dodd, T., Mihaylova, L., and Anderson, S. R. (2021). Simultaneous localization and mapping for inspection robots in water and sewer pipe networks: A review. *IEEE access*, 9:140173–140198.

Amjad, B., Ahmed, Q. Z., Lazaridis, P. I., Hafeez, M., Khan, F. A., and Zaharis, Z. D. (2023). Radio slam: A review on radio-based simultaneous localization and mapping. *IEEE Access*, 11:9260–9278.

Ashley, R. and Hopkinson, P. (2002). Sewer systems and performance indicators—-into the 21st century. *Urban water*, 4(2):123–135.

Balaguer, C. and Abderrahim, M. (2008). *Robotics and automation in construction*. BoD–Books on Demand.

Bandala, M., West, C., Monk, S., Montazeri, A., and Taylor, C. J. (2019). Vision-based assisted tele-operation of a dual-arm hydraulically actuated robot for pipe cutting and grasping in nuclear environments. *Robotics*, 8(2):42.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pages 404–417. Springer.

Bayraktar, E. and Boyraz, P. (2017). Analysis of feature detector and descriptor combinations with a localization experiment for various performance metrics. *Turkish Journal of Electrical Engineering and Computer Sciences*, 25(3):2444–2454.

Blochliger, F., Fehr, M., Dymczyk, M., Schneider, T., and Siegwart, R. (2018). Topomap: Topological mapping and navigation based on visual slam maps. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3818–3825. IEEE.

Boudjabi, S., Ferreira, A., and Krupa, A. (2003). Modeling and vision-based control of a micro catheter head for teleoperated in-pipe inspection. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 3, pages 4282–4287. IEEE.

Brunskill, E., Kollar, T., and Roy, N. (2007). Topological mapping using spectral clustering and classification. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3491–3496. IEEE.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332.

Chataigner, F., Cavestany, P., Soler, M., Rizzo, C., Gonzalez, J.-P., Bosch, C., Gibert, J., Torrente, A., Gomez, R., and Serrano, D. (2019). Arsi: an aerial robot for sewer inspection. In *Advances in Robotics Research: From Lab to Market: ECHORD++: Robotic Science Supporting Innovation*, pages 249–274. Springer.

Chen, G., Pan, L., Xu, P., Wang, Z., Wu, P., Ji, J., Chen, X., et al. (2020). Robot navigation with map-based deep reinforcement learning. In *2020 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, pages 1–6. IEEE.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848.

Cherabier, I., Schonberger, J. L., Oswald, M. R., Pollefeys, M., and Geiger, A. (2018). Learning priors for semantic 3d reconstruction. In *Proceedings of the European conference on computer vision (ECCV)*, pages 314–330.

Choset, H. and Nagatani, K. (2001). Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization. *IEEE Transactions on robotics and automation*, 17(2):125–137.

Chung, S.-H., Hong, T.-H., Han, S.-W., Son, J.-H., and Lee, S.-Y. (2006). Life cycle cost analysis based optimal maintenance and rehabilitation for underground infrastructure management. *KSCE Journal of civil engineering*, 10:243–253.

Dellaert, F., Fox, D., Burgard, W., and Thrun, S. (1999). Monte carlo localization for mobile robots. In *Proceedings 1999 IEEE international conference on robotics and automation (Cat. No. 99CH36288C)*, volume 2, pages 1322–1328. IEEE.

Deng, C., Qiu, K., Xiong, R., and Zhou, C. (2019). Comparative study of deep learning based features in slam. In *2019 4th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, pages 250–254. IEEE.

Diankov, R. and Kuffner, J. (2008). Openrave: A planning architecture for autonomous robotics. *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34*, 79.

Doychinov, V., Zaidi, S., Chudpooti, N., Akkaraekthalin, P., Somjit, N., Robertson, I., Horoshenkov, K., and Wakeling, J. (2021). A study of wireless communications systems for robotic communications in underground pipes and ducts. In *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 209–213. IEEE.

Doyle, M. W., Stanley, E. H., Havlick, D. G., Kaiser, M. J., Steinbach, G., Graf, W. L., Galloway, G. E., and Riggsbee, J. A. (2008). Aging infrastructure and ecosystem restoration. *Science*, 319(5861):286–287.

Durrant-Whyte, H. and Bailey, T. (2006). Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110.

Eiger, G., Shamir, U., and Ben-Tal, A. (1994). Optimal design of water distribution networks. *Water resources research*, 30(9):2637–2646.

Engel, J., Schöps, T., and Cremers, D. (2014). Lsd-slam: Large-scale direct monocular slam. In *European conference on computer vision*, pages 834–849. Springer.

Fox, D., Thrun, S., Burgard, W., and Dellaert, F. (2001). Particle filters for mobile robot localization. In *Sequential Monte Carlo methods in practice*, pages 401–428. Springer.

Gabriel, M. (2024). The big turn-off: shutting down the gas grid. Accessed: 2024-08-19.

Garcia-Fidalgo, E. and Ortiz, A. (2015). Vision-based topological mapping and localization methods: A survey. *Robotics and Autonomous Systems*, 64:1–20.

Golubchikov, O. and Thornbush, M. (2020). Artificial intelligence and robotics in smart city strategies and planned smart development. *Smart Cities*, 3(4).

Graves, A. and Graves, A. (2012). Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45.

Gurguze, G. and Turkoglu, I. (2018). Energy management techniques in mobile robots. *International Journal of Energy and Power Engineering*, 11(10):1085–1093.

Hao, T., Rogers, C., Metje, N., Chapman, D., Muggleton, J., Foo, K., Wang, P., Pennock, S. R., Atkins, P., Swingler, S., et al. (2012). Condition assessment of the buried utility service infrastructure. *Tunnelling and Underground Space Technology*, 28:331–344.

Herrán-González, A., De La Cruz, J. M., De Andrés-Toro, B., and Risco-Martín, J. L. (2009). Modeling and simulation of a gas distribution pipeline network. *Applied Mathematical Modelling*, 33(3):1584–1600.

Hu, H., Fang, H., Wang, N., Liu, H., Lei, J., Ma, D., and Dong, J. (2022). A study of automatic recognition and localization of pipeline for ground penetrating radar based on deep learning. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5.

Jahanshahi, M. R., Shen, W.-M., Mondal, T. G., Abdelbarr, M., Masri, S. F., and Qidwai, U. A. (2017). Reconfigurable swarm robots for structural health monitoring: a brief review. *International Journal of Intelligent Robotics and Applications*, 1:287–305.

Jang, H., Kim, T. Y., Lee, Y. C., Song, Y. H., and Choi, H. R. (2022). Autonomous navigation of in-pipe inspection robot using contact sensor modules. *IEEE/ASME Transactions on Mechatronics*, 27(6):4665–4674.

Jensfelt, P. (2001). *Approaches to mobile robot localization in indoor environments.* PhD thesis, KTH.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.

Kanayama, Y., Kimura, Y., Miyazaki, F., and Noguchi, T. (1990). A stable tracking control method for an autonomous mobile robot. In *Proceedings., IEEE International Conference on Robotics and Automation*, pages 384–389. IEEE.

Karimanzira, D. (2023). Simultaneous pipe leak detection and localization using attention-based deep learning autoencoder. *Electronics*, 12(22):4665.

Kiumarsi, B., Vamvoudakis, K. G., Modares, H., and Lewis, F. L. (2017). Optimal and autonomous control using reinforcement learning: A survey. *IEEE transactions on neural networks and learning systems*, 29(6):2042–2062.

Koenig, N. and Howard, A. (2004). Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 2149–2154. IEEE.

Kortenkamp, D. and Weymouth, T. (1994). Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the Twelfth AAAI National Conference on Artificial Intelligence*, pages 979–984.

Kreković, M., Dokmanić, I., and Vetterli, M. (2016). Echoslam: Simultaneous localization and mapping with acoustic echoes. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11–15. Ieee.

Lattanzi, D. and Miller, G. (2017). Review of robotic infrastructure inspection systems. *Journal of Infrastructure Systems*, 23(3):04017004.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, Y., Yu, A. W., Meng, T., Caine, B., Ngiam, J., Peng, D., Shen, J., Lu, Y., Zhou, D., Le, Q. V., et al. (2022). Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17182–17191.

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

Menendez, E., Victores, J. G., Montero, R., Martínez, S., and Balaguer, C. (2018). Tunnel structural inspection and assessment using an autonomous robotic system. *Automation in Construction*, 87:117–126.

Moser, A. P. (1990). *Buried pipe design.*

Mounce, S. R., Shepherd, W. J., Boxall, J. B., Horoshenkov, K. V., and Boyle, J. H. (2021). Autonomous robotics for water and sewer networks. *IAHR Hydrolink*, 2:55–62.

Mur-Artal, R., Montiel, J. M. M., and Tardos, J. D. (2015). Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163.

Narayanan, L. K. and Sankaranarayanan, S. (2019). Iot enabled smart water distribution and underground pipe health monitoring architecture for smart cities. In *2019 IEEE 5th international conference for convergence in technology (I2CT)*, pages 1–7. IEEE.

Naseer, T., Spinello, L., Burgard, W., and Stachniss, C. (2014). Robust visual robot localization across seasons using network flows. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28.

Nassiraei, A. A., Kawamura, Y., Ahrary, A., Mikuriya, Y., and Ishii, K. (2007). Concept and design of a fully autonomous sewer pipe inspection mobile robot" kantaro". In *Proceedings 2007 IEEE international conference on robotics and automation*, pages 136–143. IEEE.

Nguyen, T., Blight, A., Pickering, A., Jackson-Mills, G., Barber, A., Boyle, J., Richardson, R., Dogar, M., and Cohen, N. (2022). Autonomous control for miniaturized mobile robots in unknown pipe networks. *Frontiers in Robotics and AI*, 9:997415.

Olson, E. B. (2009). Real-time correlative scan matching. In *2009 IEEE International Conference on Robotics and Automation*, pages 4387–4393. IEEE.

Pandey, A., Pandey, S., and Parhi, D. (2017). Mobile robot navigation and obstacle avoidance techniques: A review. *Int Rob Auto J*, 2(3):00022.

Parker, J. R. (2010). *Algorithms for image processing and computer vision.* John Wiley & Sons.

Parrott, C., Dodd, T. J., Boxall, J., and Horoshenkov, K. (2020). Simulation of the behavior of biologically-inspired swarm robots for the autonomous inspection of buried pipes. *Tunnelling and Underground Space Technology*, 101:103356.

Pipeline and Journal, G. (2024). National grid studying uk hydrogen pipeline network potential. *Pipeline and Gas Journal.* Accessed: 2024-08-19.

Pomerleau, F., Colas, F., Siegwart, R., et al. (2015). A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends® in Robotics*, 4(1):1–104.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A. Y., et al. (2009). ROS: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan.

Rao, Y. R., Prathapani, N., and Nagabhooshanam, E. (2014). Application of normalized cross correlation to image registration. *International Journal of Research in Engineering and Technology*, 3(5):12–16.

Repair, P. (2024). Water supply in the uk - 5 facts about our water and the network behind it. Accessed: 2024-08-19.

Resources, G. (2020). New digital map of underground pipes and cables for the united kingdom. Accessed: 2024-08-19.

Rezende, A. M., Júnior, G. P., Fernandes, R., Miranda, V. R., Azpúrua, H., Pessin, G., and Freitas, G. M. (2020). Indoor localization and navigation control strategies for a mobile robot designed to inspect confined environments. In *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, pages 1427–1433. IEEE.

Roh, S.-g. and Choi, H. R. (2005). Differential-drive in-pipe robot for moving inside urban gas pipelines. *IEEE transactions on robotics*, 21(1):1–17.

Roumeliotis, S. I. and Bekey, G. A. (2000). Bayesian estimation and kalman filtering: A unified framework for mobile robot localization. In *Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065)*, volume 3, pages 2985–2992. IEEE.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee.

Rugg-Gunn, D. and Aitken, J. M. (2022). Investigating scene visibility estimation within orb-slam3. In *Annual Conference Towards Autonomous Robotic Systems*, pages 155–165. Springer.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Sharifi, K., Sabeti, M., Rafiei, M., Mohammadi, A. H., and Shirazi, L. (2018). Computational fluid dynamics (cfd) technique to study the effects of helical wire inserts on heat transfer and pressure drop in a double pipe heat exchanger. *Applied Thermal Engineering*, 128:898–910.

Song, T., Zhang, X., Ding, M., Rodriguez-Paton, A., Wang, S., and Wang, G. (2022). Deepfusion: A deep learning based multi-scale feature fusion method for predicting drug-target interactions. *Methods*, 204:269–277.

Su, Z., Zhou, X., Cheng, T., Zhang, H., Xu, B., and Chen, W. (2017). Global localization of a mobile robot using lidar and visual features. In *2017 IEEE international conference on robotics and biomimetics (ROBIO)*, pages 2377–2383. IEEE.

Tan, W., Liu, H., Dong, Z., Zhang, G., and Bao, H. (2013). Robust monocular slam in dynamic environments. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 209–218. IEEE.

Team, P. A. (2019). Position paper: Intelligent buried pipe infrastructure: Present and future. Technical report, Pipebots Project. Available at: `https://pipebots.ac.uk/publications/`.

Thrun, S. (1998a). Bayesian landmark learning for mobile robot localization. *Machine learning*, 33:41–76.

Thrun, S. (1998b). Learning metric-topological maps for indoor mobile robot navigation. *Artificial intelligence*, 99(1):21–71.

Thrun, S. (2002). Probabilistic robotics. *Communications of the ACM*, 45(3):52–57.

Thrun, S., Burgard, W., and Fox, D. (2000). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 1, pages 321–328. IEEE.

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

Vrontis, D., Christofi, M., Pereira, V., Tarba, S., Makrides, A., and Trichina, E. (2023). Artificial intelligence, robotics, advanced technologies and human resource management: a systematic review. *Artificial Intelligence and International HRM*, pages 172–201.

Wang, M., Hu, D., Chen, J., and Li, S. (2023). Underground infrastructure detection and localization using deep learning enabled radargram inversion and vision based mapping. *Automation in Construction*, 154:105004.

Wang, S., Clark, R., Wen, H., and Trigoni, N. (2017). Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2043–2050. IEEE.

Wang, X., Luo, L., Xiang, J., Zheng, S., Shittu, S., Wang, Z., and Zhao, X. (2021). A comprehensive review on the application of nanofluid in heat pipe based on the machine learning: Theory, application and prediction. *Renewable and Sustainable Energy Reviews*, 150:111434.

Wong, C., Yang, E., Yan, X.-T., and Gu, D. (2018). Autonomous robots for harsh environments: a holistic overview of current solutions and ongoing challenges. *Systems Science & Control Engineering*, 6(1):213–219.

Yu, Y., Worley, R., Anderson, S., and Horoshenkov, K. V. (2023). Microphone array analysis for simultaneous condition detection, localization, and classification in a pipe. *The Journal of the Acoustical Society of America*, 153(1):367–383.

Zhang, R., Worley, R., Edwards, S., Aitken, J., Anderson, S., and Mihaylova, L. (2023). Visual simultaneous localization and mapping for sewer pipe networks leveraging cylindrical regularity. *IEEE Robotics and Automation Letters*, 8(6):3406–3413.

Zhang, Z., Yu, J., Tang, J., Xu, Y., and Wang, Y. (2022). Mr-topomap: Multi-robot exploration based on topological map in communication restricted environment. *IEEE Robotics and Automation Letters*, 7(4):10794–10801.

Zhao, F., Huang, Q., and Gao, W. (2006). Image matching by normalized cross-correlation. In *2006 IEEE international conference on acoustics speech and signal processing proceedings*, volume 2, pages II–II. IEEE.

Zhao, J., Gao, J., Zhao, F., and Liu, Y. (2017). A search-and-rescue robot system for remotely sensing the underground coal mine environment. *Sensors*, 17(10):2426.

Zhou, C., Zhang, H., Shen, X., and Jia, J. (2017). Unsupervised learning of stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1567–1575.