Delft University of Technology

FACULTY MECHANICAL, MARITIME AND MATERIALS ENGINEERING
Department Marine and Transport Technology

Mekelweg 2
2628 CD Delft
the Netherlands
Phone +31 (0)15-2782889
Fax +31 (0)15-2781397
www.mtt.tudelft.nl

Specialisation:	Transport Engineering and Logistics
Report number:	2020.MME.8471
Title:	**Collision Avoidance for Autonomous Inland Vessels using Stereovision**
Author:	M. Hepworth

Assignment:	Master Thesis (ME54035)
Confidential:	No
Initiator (university):	Prof. dr. R. R. Negenborn
Initiator (company):	N/A
Supervisor:	Dr. V. Reppa
Date:	January 1, 2021

Master Thesis Project

Collision Avoidance for Autonomous Inland Vessels using Stereovision

M. Hepworth

TU Delft

Master Thesis Project

Collision Avoidance for Autonomous Inland Vessels using Stereovision

by

M. Hepworth

submitted in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at Delft University of Technology,
to be defended publicly on the 14th January, 2020 at 10:00 AM.

Student Number: 4927648
MSc Track: Transport Engineering and Logistics
Report Number: 2020.MME.8471

Thesis committee:	Prof. dr. R. R. Negenborn,	TU Delft, Committee Chair
	Dr. V. Reppa,	TU Delft, Academic Supervisor
	Ir. V. Garofano,	TU Delft, Daily Supervisor
	Dr. ir. P.R. Wellens,	TU Delft, External Committee Member
Date:	January 2021	

TU Delft

Preface

I would like to express my gratitude to my supervisors and committee members Ir. Vittorio Garofano, Dr. Vasso Reppa and Prof.dr. Rudy Negenborn for their feedback, guidance and support throughout my thesis. This gratitude is further extended to the Researchlab Autonomous Shipping for facilitating my project and my fellow colleagues Bart, Maurits and Wouter, it has been a pleasure working with you.

A thank you is also of course owed to all my family and friends, whose support throughout the duration of my studies was essential to my success. I could not have done it without you and sincerely hope that at this, the final stage of my journey to becoming an engineer to have done you all proud.

M. Hepworth
Delft, January 2021

Executive Summary

The inland waterway once enabled an industrial revolution, yet the emergence of coalescent road networks has seen its true worth be all but disregarded. Despite the amenity of unimodal travel being compelling, growing awareness for sustainability has reignited interest in more fuel efficient modalities for transportation. Autonomous shipping has the potential to increase efficiency, reliability and safety and will arguably play a major role in the evolving transport revolution, returning the transport modality to its former glory.

The main objective of this research is to cater a collision avoidance strategy to the inland waterway through the development of tailored Guidance and Navigation Systems. An approach to local path planning is introduced to handle the challenges of collision avoidance on the inland waterway and a better understanding of the primary role that stereovision sensors could assume in enabling inland autonomy is gained. Achieving this objective requires first a reflection upon existing work through a study into the state-of-the-art. Subsequently, the Guidance and Navigation Systems are developed and implemented on a scale test vessel. Finally experimental testing is conducted for the evaluation of the developed system performance.

Literature Study

Maritime autonomy has been widely researched, however a focus has only recently emerged on inland applications and the specific challenges the environment presents. It was found that a number of sensors are typically utilised onboard autonomous vessels, particularly inland vessels. The closer interactions between vessels in this environment requires low-mid range sensor coverage. Whilst stereovision is one of the sensor types utilised, it takes a back seat role with LiDAR remaining the favoured choice. Despite this, it was discovered that autonomous cars have developed a preference for vision based technology as a primary source of sensor data. The automotive industry has turned to multi-device setups to achieve sufficient coverage using vision sensors, dropping the dependency on the expensive, unsightly LiDAR sensor technology. A gap was identified in inland autonomous research to this regard, with the possibility that vision sensors could provide a lucrative research direction to the field in the same way the technology has assisted in the evolution of autonomous ground vehicles.

A number of collision avoidance approaches were identified in the literature for autonomous vessels, with open-water applications being particularly well covered. Inland applications however have not received the same research focus. Furthermore, the restricted channels synonymous with the inland waterway hinder the direct application of existing open-water techniques for conflict detection and resolution. An additional gap was highlighted here, whereby further research was required to tailor a collision avoidance procedure to the inland waterway environment. The importance of experimental testing was also highlighted in the review and a number of suitable key-performance-indicators were selected to assist in the evaluation of Navigation and Guidance System performance.

Navigation and Guidance

The Navigation System developed is based around a multi-device stereovision setup to provide low-mid range perception and localisation onboard autonomous inland vessels. Four perception devices positioned around the vessel enable a constant lookout to be maintained. The utilisation a convolutional neural network to conduct object detection has demonstrated its suitability to the autonomous inland vessel application. Not only does this approach provide obvious advantages over arbitrary object detection but does so at impressive inference rates, even when computationally limited to a small form-factor computer onboard a scale vessel. A refined approach to the acquisition of stereovision depth data avoids the unnecessary post-processing and manipulation of dense point clouds. The utilisation of a Visual Inertial Odometry approach provides localisation using only stereovision and IMU data. The offloading of the task to a singular compact device, allows a significant boost to small scale applications and yet the VIO approach also lends itself well to larger application setups. Secondary processing of the perception and localisation data through a mapping procedure provides a coalescent output for the Navigation System, specifically configured to provide pertinent information for onward collision avoidance procedures.

The Guidance System incorporates a collision avoidance protocol capable of handling the specific challenges of the inland waterway. A new configuration for conflict detection regions has been proposed to overcome the issues that arise from the inland waterway architecture and the subsequent nature of the interactions that take place. The approach sees the use of rectangular vessel domain regions whose dimensions are influenced by a function of the vessel speed. The collision resolution approach integrates collision regulations and a secondary emergency procedure within a rule-based implementation compatible with the data available from the stereovision based Navigation System. This two tier system prevents the AV from blindly following the collision regulations when a contact vessel is recognised to be non-compliant or when an extremely close encounter requires additional action to be taken. The local path planner incorporates a roll-out-trajectory generation technique that has been adapted and optimised for application on an inland autonomous vessel.

The Guidance and Navigation systems have been implemented in a manner to promote open-source accessibility and software standardisation. This has been done with a view towards creating a strong foundation for ongoing research developments in inland autonomy. The implementation utilises the ROS middleware framework, which provides a standardised communications protocol and package-based structure. Programming follows a consistent approach, with Python being used throughout and a modular sub-structure creates an ideal framework for future developments and potential application to other vessels.

Experimental Evaluation

The Navigation System was independently evaluated through experimental testing of the perception and localisation tasks to provide a qualitative and quantitative indication of attainable performance. The perception solution was evaluated with a focus upon the robustness of obstacle detection and the accuracy of positioning. Under the defined precision/recall key performance indicators, the obstacle detection procedure was found to be capable of maintaining high precision with increasing recall, indicating good detection performance. However the significance of this result is limited to the specific experimental environment. The accuracy of obstacle positioning was found to be reasonable with a root mean square position error of 0.31 metres being found.

Experimental testing to evaluate localisation performance was conducted using two path types, a slalom/zigzag style path and a straight path. Along a 15m length, the position error metric was found to be 0.129m and the heading error to be 5.3°. A variation in localisation performance was however demonstrated along the path length, with the performance decreasing with an increase in length. These quantitative evaluations were made under normal operational cases, however the localisation system was also observed on multiple occasions to be susceptible to the impact of the kidnapped robot problem.

Collision avoidance experiments provided an evaluation of the combined Guidance and Navigation System performance. Across all of the thirty tests conducted, an average avoidance success rate of 90% was achieved, meaning that 10% of interactions resulted in collision due to insufficient action. Of the successful tests however, only 40 out of the total 90% percent saw fully compliant avoidance with the remainder seeing either the premature activation of emergency procedure or the incorrect COLREG mode being activated.

The experimental testing highlighted that whilst the Guidance System performed exceptionally well under simulated testing, with real world data supplied from the stereovision devices this same performance could not be accomplished. The weakest link in the entire network during this combined testing appeared to be the localisation system which was prone to sudden failures. It is concluded that further developments are required to improve Navigation System performance and/or tune the collision avoidance parameters better to suit the quality of data available from the sensors.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

AIS	Automatic Identification System
AV	Autonomous Vessel
COLREG	International Regulations for Preventing Collisions at Sea
COP	Closest Obstacle Point - *closest detected point of an obstacle*
CV	Contact Vessel - *an obstacle of vessel classification status with which a close quarter interaction is occurring*
GNC	Guidance Navigation and Control System
GPS	Global Positioning System
IMU	Inertial Measurement Unit
IOU	Intersection over Union
KPI	Key Performance Indicator
MOOS	Mission Orientated Operating Suite
MTT	Department of Maritime and Transport Technology
RAS	Researchlab Autonomous Shipping
ROI	Region of Interest
ROS	Robot Operating System
RTG	Roll-out Trajectory Generation
SLAM	Simultaneous Localisation and Mapping
UAV	Unmanned Aerial Vehicle
USV	Unmanned Surface Vehicle
VIO	Visual Inertial Optometry

Greek Symbols

α	Approach Angle (°) – *the direction of the CV velocity vector relative to the AV heading.*
β	Relative Bearing (°) – *the angle between the AVs forward direction and CPA of an obstacle.*
$\epsilon_{1,2,3}$	Imaginary components of quaternion vector
η	Real component of quaternion vector
γ	Angle of camera orientation (°)
ϕ	Roll (rad)
ψ	Yaw (rad)
τ	Track Angle (°) – *the direction of the CV velocity vector in the global coordinate frame.*
θ	Pitch (rad)
φ	Heading (°) – *the orientation of the AV in the global coordinate frame.*

Latin Symbols

B	Vessel Beam (m)- *Width of a vessel at its widest point.*
c_i	Integer COLREG Weight
d	Distance to the edge of the waterway from the current path (m)
d_i	Integer Distance Weight
P_x	Proximity between a Contact Vessel and candidate trajectory in the y-axis
q	Quaternion Vector
r_i	Integer Collision Risk Weight
t	Continuous time (s)
u	Speed of the Autonomous Vessel

Subscripts

$_a$	Aft - *At or toward the rear of a vessel.*
$_b$	Body-fixed frame
$_f$	Fore - *At or toward the front of a vessel.*
$_p$	Port - *The left side of a vessel.*
$_s$	Starboard - *The right side of a vessel.*
$_{AV_D}$	Autonomous Vessel Domain
$_{av}$	Autonomous Vessel
$_{CO}$	Origin of body-fixed frame
$_{cv}$	Contact Vessel

1 Introduction

The inland waterway once enabled an industrial revolution, yet the emergence of coalescent road networks has seen its true worth be all but disregarded. Despite the amenity of unimodal travel being compelling, growing awareness for sustainability has reignited interest in more fuel efficient modalities for transportation. Traffic congestion remains a persistent issue in urban areas and logistical hubs and fixed infrastructure links do not always offer expedient options to cross waterways. Increased utilisation of the inland waterways could offer a solution to achieving emissions targets, easing congestion and providing alternative crossing solutions. Attaining said utilisation and future-proofing the market share of the inland waterway does however require innovative solutions.

Concepts such as synchromodality have presented approaches to enable a modal shift in hinterland freight transportation through the creation of an interconnected, integrated and cooperative freight transportation network [1]. However its adaptive nature demands high efficiency and reliability from the inland waterway network. Autonomous shipping has the potential to increase efficiency and reliability and will arguably play a major role in the evolving transport revolution, returning the transport modality to its former glory.

When reviewing literature, one encounters a whole host of terms used to describe vessels with various levels of automation and determining the most suitable term for onward use presents a challenge. The term Unmanned Surface Vehicle (USV) is perhaps most prevalent, however it can refer to both a vessel that navigates independently and a vessel that is controlled remotely by a human operator, it only states that no crew are required onboard. The terms Autonomous Surface Vehicle (ASV) and Maritime Autonomous Surface Ship (MASS) are also encountered which appear to be adaptions of USV only placing emphasis towards autonomy. These terms are encountered in literature describing vessels with various levels of autonomy and there remains some debate as to the exact definition of autonomous and its relation to automated [2].

For clarity, this thesis will consider autonomous to describe full automation whereby operation requires neither human decision making, nor intervention. This research will focus on the application of autonomy tailored for surface vehicles on the inland waterway environment and will refer to said craft throughout as an Autonomous Vessel (AV). The author refrains from using the term unmanned as some potential passenger applications may still require crew for stewarding tasks in the initial adoption stage. Whist numerous tasks may require on board staff, presence would primarily be to assume responsibility for passenger safety in emergency scenarios, particularly during the transitional stages of autonomy.

It would be fair to conclude from the wealth of literature that autonomous vessels have been a topic of interest over the past two decades. The research conducted has shown significant advancements, yet there still remains incentive for further studies. This particular study will aim its focus towards the application of autonomy to inland vessels where despite recent attention, there still remains significant gaps for further research. Prior to divulging the scope of this thesis, it seems prudent to first detail the primary incentives which motivate research into autonomous inland vessels.

Ports and hinterland networks have recently turned focus away from raw capacity incentives towards the pursuit of efficiency, a switch that can partially be attributed to sustainability objectives. With road transportation being the dominant modality for hinterland logistics, the more sustainable alternatives of barge and rail transport are now starting to have a growing appeal. The Port of Rotterdam Authority for instance have set an objective to reduce onward hinterland transport by road to 35% by 2035, a reduction of 20% from the 2010 figure [3]. Autonomy on the inland waterway could help in securing this objective due to anticipated performance improvements through the close adherence to instruction.

Beyond freight transportation, inland waterways also provide an option for alleviating passenger transport from road traffic networks. Ferries, water busses and water taxis all represent alternative modalities, which have the capacity to reduce road congestion and travel times. In some cases even providing an alternative to hard infrastructure where a bridge or tunnel is not feasible or affordable. Autonomy could provide the framework for a greater utilisation of these passenger services by increasing capacity and service frequency. However these smaller applications do not benefit from quite the same economy of scale as larger inland freight vessels, lessening the capacity for investment in automation. There is also a larger intrinsic risk factor to safety, as humans on board the vessel yields further complications during hazard analysis [4].

The removal of human error is frequently acknowledged as one of the key incentives of autonomy. The European inland waterway network however exhibits a rather exemplary safety record, meaning it would be unfair to say that there already exists a major incentive to this end. The increase in traffic that would arise from a modal shift however may infringe on the networks safety record. Currently the inland waterway is arguably underutilised and so a significant increase in traffic would push the network towards its capacity, especially in the bottleneck regions. The likelihood of an accident occurring increases significantly when approaching capacity limits leading to it being anticipated that the current safety record would not hold with the increased traffic. Autonomy perhaps offers a solution here as safety is one of the key factors influencing waterway capacity [5]. Autonomous vessels, particularly those sailing in cooperative fleets bare the potential to safely operate in denser waterways than their conventional counterparts. This means that current limit to capacity could be tried, increasing traffic without warranting the same safety concerns.

The consequential increase in demand from the modal shift would also lead to an anticipated shortage of appropriately skilled crew. A fully autonomous vessel could be partially or entirely unmanned, which helps bi-pass labour issues along with a host of secondary benefits. The risk of workplace injuries would be reduced and the working environment of employees significantly improved. An unmanned vessel could further allow for simplified ship designs, increasing cargo capacity and reducing manufacture costs through the removal of crew and safety facilities. Despite the higher initial investment that could be anticipated with an autonomous vessel, the ongoing costs that could be associated with paying personnel would also not prevail, sparing operating expenses in the long run.

Verberght and van Hansell [2] present a comprehensive analysis on the potential of a fully automated and unmanned vessel for inland freight transportation. The paper conducts a two-fold analysis to provide a good indication of the current state of autonomous inland vessels and the challenges that still need to be overcome. From an economic perspective, cashflow comparisons of autonomous inland vessels and their manned equivalents in various scenarios yielded some promising results. The results suggest that autonomous vessels do present a potentially viable business case on European inland waterway network. Infrastructure investments are expected to rise approximately 25% to enable autonomy in Europe (lock mooring), yet the anticipated performance superiority of autonomous vessels would level off its cost to society. On average however, it was found that an autonomous vessel only exhibited a superior business case when the conventional vessel had a crew in excess of six members. This significantly limits the financial benefit of autonomous vessels over their conventional counterparts.

Summarising, there are numerous incentives for a modal shift to the inland waterway and autonomy has the potential to enable the transfer, bringing with it sustainability and efficiency benefits. It may be argued that the current benefits of autonomy on are not overwhelming certainly not from an economic perspective, however development is still very much in its infancy. On a performance front autonomy has a number of a benefits over human based control, most notably being better informed and consistently decisive. Furthermore, should alternative modalities advance their performance through automation, the market share of the inland waterway network could diminish all together if it fails to keep up. To remain competitive, it is essential that inland vessels evolve with the times and to do so applications require an accessible means to achieve autonomy.

1.1. Problem Statement

This study aims to further the research into collision avoidance for autonomous inland vessels. It is recognised that achieving autonomous collision avoidance is reliant upon a robust Guidance, Navigation and Control system. The Control branch remains predominantly outside of the scope of this project as the existing system of the test vessel will be used for testing the Guidance and Navigation Systems with only a few minor alterations. The Guidance and Navigation branches instead shall form the main focus and the following challenges have been highlighted

Navigation solutions for perception and localisation that were encountered for inland waterways incorporate a host of sensors, which whilst useful during developmental testing could eventually prove to be a barrier to adoption. The restricted nature of the inland waterway network leads to the requirement of close range perception. For gathering near range depth perception, LiDAR is arguably the most popular choice, with stereovision playing a secondary role if any. In the automotive sector, where near range perception is also critical, imaging sensors are assuming a leading role with LiDAR being left out entirely. The motivation behind this choice is predominantly due to expense, something seldom considered in maritime research, with the exception of a low-cost radar solution for collision avoidance [6]. Sensor affordability could too play a vital role in making autonomous technology accessible to a wider range of inland vessel applications in the future. Imaging sensors can provide solutions to both perception and localisation tasks using stereovision setups, making it beneficial to study the potential that this solution can provide to the navigation sub-system of autonomous inland vessels.

It was found in the literature that many guidance solutions often converged on local optima, therefore care should be taken when designing a guidance system to ensure that the local avoidance does not greatly impact the optimality of the global plan. Existing global planners for inland waterways appear to already offer sufficient solutions, with tailored techniques having been developed to optimise for the shortest path and for fuel efficiency. Whilst global planning can undoubtedly be further optimised, local planning solutions for inland waterways require far more attention as they still lack basic functionalities. Static obstacle avoidance solutions have been implemented, however avoiding collisions with dynamic obstacles considering COLREGs remains an ongoing topic of research for inland applications. Although strategies for motion planning, conflict detection and conflict resolution can potentially be adopted from non-inland research, the selection of these techniques or adaptations of must be compatible with the inland environment and the specific challenges its restricted waterways present. Therefore, this study also aims to propose a strategy for collision avoidance that is suited to the inland waterway.

The implementation of the entire Guidance Navigation and Control system should ideally be conducted in a way to promote open-source accessibility and software standardisation. It is common during research directions to focus on the sub-system optimality of implementation, without considering the global system architecture despite this being critical to the realisation of full autonomy. Therefore, when developing the Guidance, Navigation and Control system, a modular structure is preferable with a consistent programming language and communications protocol. An open source GNC system to act as a basis for research into autonomous inland vessels would provide a significant boost to development in the field.

System evaluation is critical in understanding the performance of the sub-systems, which leads to the problem encountered. A guidance sub-system can be tested using simulations, however the navigation sub-system requires experimental testing in order to assess its performance. Furthermore when considering the real-world performance, a guidance system is only as good as the information its receives. Experimental testing of collision avoidance was found to be lacking in many studies yet generally considered of high importance. Through such testing, major bottlenecks can be highlighted to help focus future research directions and developments towards achieving inland autonomy.

1.2. Research Scope

The main objective of this research is to cater a collision avoidance strategy to the inland waterway through the development of specifically designed Guidance and Navigation Systems. Global path planning is not addressed, however a local path planning approach to handle the challenges of collision avoidance on the inland waterway is proposed. Inspiration is taken from the ever-evolving role that vision systems play in the field of automotive autonomy with research focus being applied to gaining a better understanding of the role that stereovision sensors could assume in enabling inland autonomy.

Main Question

How can collision avoidance be achieved by autonomous inland vessels using stereovision?

Research Questions

1. What is the state-of-the-art in inland autonomy and collision avoidance?
   (a) What sensors are currently used in autonomous vessel navigation systems and what potential does exist for stereovision sensors?
   (b) How is collision avoidance currently achieved by guidance systems and what are the requirements for collision avoidance on the inland waterway?
   (c) What are the most appropriate KPIs for the evaluation of (sub-)system performance?
2. How can perception and localisation tasks be achieved by a navigation system using stereovision?
   (a) How can stereovision provide mid-range perception onboard a vessel?
   (b) How can stereovision sensors be applied to accomplish localisation?
3. How can a guidance system avoid collisions with dynamic obstacles within an inland waterway environment?
   (a) How can collision conflict be detected between an autonomous inland vessel and a contact vessel?
   (b) How can a local path planner be configured to handle conflict resolution and collision avoidance in an inland environment?
4. How can the guidance and navigation modules be implemented towards software standardisation and accessibility?
5. How well do the guidance and navigation systems perform under experimental testing and what are their limitations?
   (a) How does the stereovision based perception system fair under precision/recall testing and position error KPIs?
   (b) How well does the stereovision based localisation perform under the KPIs of position and heading error?
   (c) Can the developed guidance and navigation systems enable the autonomous avoidance of collisions under various staged scenarios?

The approaches taken to address these research questions are as follows. Firstly, the state-of-the-art in inland autonomy and collision avoidance is studied through topic-focused reviews of literature and regulation. The research gaps that this thesis aims to fill shall be highlighted through this approach. Secondly, methods to achieve mid-range perception and localisation using stereovision are developed within the Navigation system using computer vision techniques tailored to the application and sensor set. Thirdly, techniques to detect and resolve collision conflict between inland vessels are proposed through the tailored application of existing maritime and automotive approaches, The approach towards implementation favours a standardised and modular structure so to ensure that the research conducted is accessible to future projects. This way the developed system can act as foundation for ongoing research developments in inland autonomy. Simulations are utilised as a means to verify and validate the collision avoidance procedure within the Guidance system. Whereas the approach to test the

Navigation system and the combined system performance, an experimental testing approach is taken. Evaluation is formally conducted using appropriate performance metrics.

1.3. Report Structure

The main body of this report is structured as seen in Table.1.1, with the research questions introduced in the previous section being addressed as indicated. The second Chapter focuses on reviewing the current state of research into inland autonomy and collision avoidance, as well as covering other areas of important background research. The report from this point, turns it attention away from knowledge expansion and towards the presentation of the conducted work. Chapter three provides an overview of the autonomous vessel, its hardware and the implemented software structure, with which answering research question four. The proposed stereovision based Navigation system is presented in detail in Chapter four following this overview, detailing specifically the implementation of perception and localisation tasks that form the focus of question two. The developed collision avoidance procedure for the inland environment is expanded upon in Chapter five which covers the Guidance system. The last research focus is covered in Chapter six where the developed systems are evaluated through addressing question five. Finally the main body of the report is closed of with conclusions being drawn and proposals for future research being made.

Chapter	**Contents**	**Questions Addressed**
1. Introduction	Problem Statement and Research Scope	-
2. Inland Autonomy and Collision Avoidance	Background research into the current state-of-the-art.	Q1
3. The Autonomous Vessel	Introduction to the Autonomous Vessel hardware and software structure	Q4
4. Stereovision Based Navigation System	The implementation of a stereovision based perception and localisation for inland autonomous vessels.	Q2
5. Collision Avoidance for an Autonomous Inland Vessel	The implementation of a collision avoidance procedure for the guidance system of an inland autonomous vessel.	Q3
6. Results	Evaluation of the developed Navigation and Guidance Systems	Q5
7. Conclusions and Recommendations	Recapitulation of conducted research, results and recommendations for future work	-

Table 1.1: Structure of Thesis Report and Research Questions Addressed

A three-part appendix can be found at the end of the report. The first of the appendices includes a scientific research paper summarising the thesis project work. The second and third of these appendices supplements the results section of the main body by expanding upon results gained through simulation and experimental testing respectively.

2

Inland Autonomy and Collision Avoidance

Most of the research into autonomy for maritime surface vehicles has been conducted with an arbitrary view towards application or with focus on sea going vessels. However this thesis is not alone in its vision of autonomous vessels on inland waterways. Recent developments by researchers in Belgium [7] have focused directly on autonomous inland barges and the Roboat research initiative [8] is exploring the use of autonomous vessels for transporting goods and people on Amsterdam's dense narrow waterways. Nonetheless, whilst of particular interest, this review will not be limited to the advancements in inland autonomy. Rather it will include findings from all maritime applications and research from the generic robotics community.

This chapter shall answer the first four research question concerning the current state of research alongside background study into applicable regulation and methodology. The specific question *What is the state-of-the-art in inland autonomy and collision avoidance?* and the four sub-questions are listed below, along with a link to the relevant section which addresses them.

1a What sensors are currently used in autonomous vessel navigation systems and how ? (Section 2.2)

1b How is collision avoidance currently achieved by guidance systems and what are the requirements for collision avoidance on the inland waterway? (Section 2.3 & 2.4)

1c What are the most appropriate KPIs for the evaluation of (sub-)system performance? (Section 2.5)

2.1. Guidance Navigation and Control System

An autonomous vehicle needs a Guidance, Navigation and Control System (GNC) to assume responsibility in the absence of the usual human operator. Figure 2.1 provides a breakdown of the system architecture and details the sub-tasks of each branch.

Control is managed by an autopilot system which determines and executes suitable actuation based upon the references received from the guidance branch. Whilst imperative for the successful execution of collision avoidance, control will not be investigated further in this study. Attention will instead be focused upon guidance which determines suitable routes and the navigation sub-tasks which supply the information for this decision making process.

2.2. Navigation

The navigation branch of a GNC is responsible for establishing the current state of the vessel and its environment. Perception and localisation tasks must provide coherent data to the guidance and control subsystems to ensure successful collision avoidance be achieved. Subsequently, suitable sensor selection is imperative.

Localisation concerns the determination of the vessel's own current location. This location is particularly important for the guidance tasks when generating paths as well as for the control system when generating reference headings and correcting path deviation errors. In the absence of accurate localisation, an autonomous vessel will struggle to remain on its path.

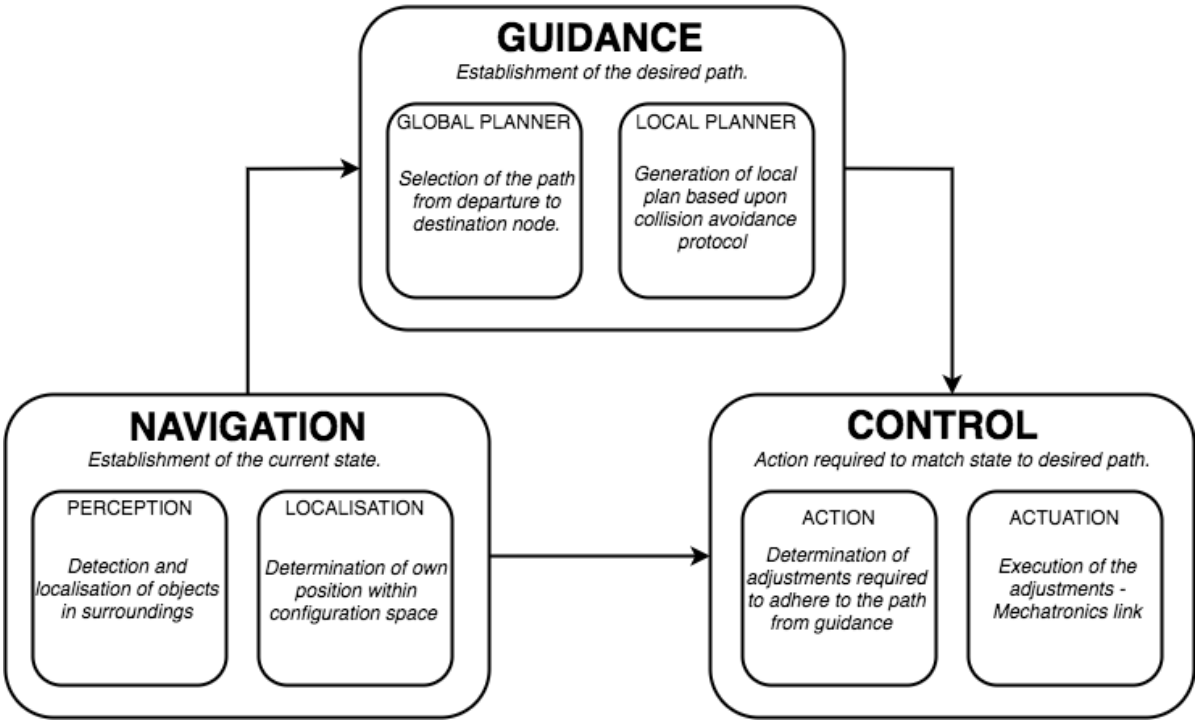


Figure 2.1: Typical components of a Global Navigation System

In maritime applications, localisation is predominately achieved using satellite navigation (GPS) and Inertial Measurement Units (IMU). GPS accuracy is typically sufficient for full size vessels however this accuracy does not scale for smaller USVs and model vessels, furthermore it is not suitable for indoor testing environments. IMUs play an important role in supplying measured angular velocities and linear accelerations which help track movement and orientation. IMU data is frequently combined with GPS data to provide more accurate localisation, particularly regarding orientation.

Simultaneous Localisation and Mapping (SLAM) algorithms can also be used for localisation as they determine pose within the configuration space. SLAM combines perception data from LiDAR and/or stereovision devices with the data from inertial measurement units (IMU). Unlike GPS, this localisation approach does lend itself to smaller, indoor applications. Localisation using SLAM can provide a higher level of precision than GPS, making it appealing for inland waterways, however robustness can usually only be achieved when using a combination of sensors and Kalman filtration to improve accuracy of positioning [8].

Perception involves the detection of obstacles and consequently determination of obstacle position and state. A key requirement in perception is the ability to measure depth as this additional dimension is critical for understanding the environment. There are numerous sensor variants which integrate the measurement of depth. Table 4.2 provides a summary of the most frequently utilised sensors and some key performance indicators.

Sensor	**Range**	**Data Frequency**	**Light & Visibility Sensitivity**	**Cost**	**Error/noise Susceptibility**
Radar	High	Low	Low	Mid	Water reflectivity error
LiDAR	Mid	Mid	High	High	Vertical noise
Imaging	Low-Mid	High	High	Low	Non-linear range error
Sonar	Low-Mid	Mid	N/A	Mid	Near-surface noise

Table 2.1: Comparison of Perception Sensors

Radar is perhaps the primary choice for perception tasks due to its proven, dependable reputation from decades of being used as a navigation aid. In terms of sea focused applications, radar is undoubtedly the most popular sensor selection for autonomous vessels. Radar range and frequency is more than sufficient for open-water environments as interactions rarely occur in close quarters. Inland vessels however tend to be subject to closer interaction between vessels, meaning that they have a greater need for close proximity depth and an increase in data frequency. In Table 2.2 this is highlighted by the disparity in sensor selections between sea and inland applications.

Inland applications tend towards sensors which provide near-range perception. LiDAR is typically the go to solution as it provides dense 3D point clouds of the nearby surroundings at a reasonable frame rate. LiDAR works by emitting lasers and recording the time of flight until the beam returns, indicating distance. Whilst being subject to noise, the sensor is only susceptible to a manageable linear error and can cover a good range. The limitation of LiDAR is primarily due to its cost especially when a 3D sensor is used, which is highly preferable for a vessel application which has freedom in pitch and roll. Further limitations of LiDAR are its longevity due to moving components which can lead to mechanical failures and its measurement sensitivity to motion and vibrations.

Imaging sensors offer another means of collecting depth data through stereovision, which works by manipulating data from two cameras located at a fixed distance from one another called the baseline. The overlapping region between the captured frames can be assessed using disparity to interpret depth. An obvious advantage of this choice is its two-dimensional field-of-view which offers 3D perception, albeit without a 360° horizontal coverage. Range is also limited as stereovision is subject to a quadratic depth error meaning that reliable perception is limited to a region determined by the baseline of the sensors. The cost of imaging sensors is however low in comparison to the other sensor options and the monocular image data can also prove very useful in supporting perception tasks such as object detection.

Whilst sonar sensors are to be expected to be encountered in underwater vehicles, they are also occasionally used by surface vehicles for perception and localisation tasks. Use in perception is however limited due to the challenges of automatic object detection and surface noise proving a major obstacle in identifying vessels with a low draft. Scanning the waterway bed can however prove useful as the sonar data can be run through a SLAM algorithm to localise the vessel in its environment.

Citation	Application	Sensors	Avoidance	COLREGs	Verification
[7]	Inland	GPS, IMU, LiDAR, Imaging	Interval Programming	No	Experiment
[8]	Inland	IMU, LiDAR, Imaging	Rollout Trajectory Generation	No	Experiment
[9]	Sea	Radar	Velocity Obstacle, Re-planning	Yes	Simulation
[10]	Sea	Radar	Re-planning	Yes	Simulation
[11]	Sea	Radar	Velocity Obstacle	Yes	Simulation
[12]	Sea	Radar, Imaging	Model Predictive Control	Yes	Both
[13]	Sea	GPS, Radar, Sonar, Imaging	Collision Cone	No	Simulation
[14]	Sea	Radar	Velocity Obstacle	No	Experimental
[15]	Sea	Radar, GPS	Artificial Potential Field	Yes	Simulation
[16]	Sea	Radar	Dynamic Window	No	Experimental

Table 2.2: Typical Maritime Collision Avoidance Systems

Autonomous Inland Vessels perhaps exhibit perception requirements closer to those of autonomous ground vehicles as opposed to their sea-based counterparts. In the automotive industry, LiDAR has also been the most popular sensor choice for achieving near range perception and was long considered a necessity. However of late some autonomous cars have deviated from this preference. Perhaps the most infamous enablers of autonomous travel, Tesla, opts for imaging techniques. A study by researchers at Cornell University has shown the capability of stereovision to conduct the tasks previously assumed only possible using LiDAR [17].

The inland systems in Table 2.2 both utilise LiDAR and stereovision, however in both cases stereovision takes a back seat role. Roboat [8] uses stereovision as a secondary input for localisation tasks and the work of Slaets et al. [7] supports LiDAR data with stereovision data to complement the forward facing point cloud. The latter study also uses the monocular frames from the individual stereo cameras to assist in perception sub-tasks such as object detection.

The emerging choice of imaging sensors over LiDAR in the automotive industry comes primarily due to the cost of the sensor, with the former providing a far more affordable solution. Imaging based perception relies heavily on intelligent computer vision techniques to enable functionality and require a multi-sensor setup in order to provide the 360° horizontal field of view that is standard with LiDAR. Secondary perception sensors are also required in poor visibility and very close quarters conditions. Tesla for example use the sensor setup in Figure 2.2 to achieve perception. Whilst inland vessels are arguably not restricted by the same necessity for consumer accessibility, reducing the initial investment cost has the potential to catalyse adoption.

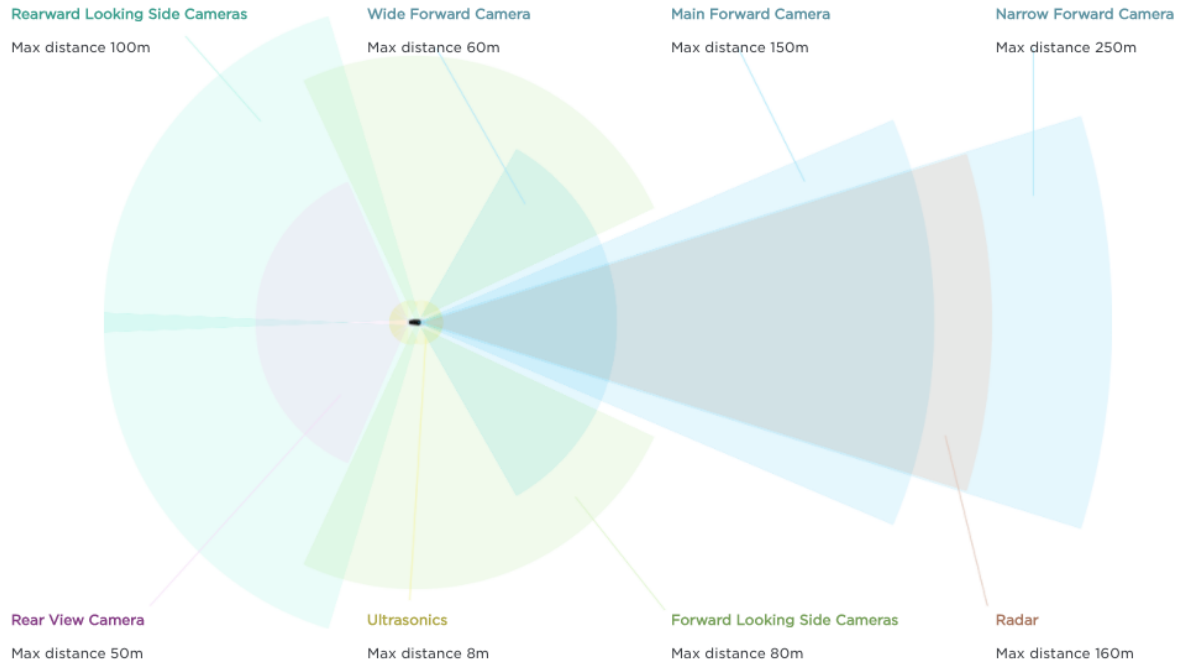


Figure 2.2: Sensor setup for Tesla Autopilot [18]

2.3. Guidance

Global planning for autonomous vessels typically involves the use of algorithms which select the shortest path to be optimal. Adapted versions of the well-known A* search algorithm are frequently implemented to carry out this task [10][19][20]. One proposal worthy of particular mention is that of Chen et al.[21] which sees the adaption of the A* algorithm specifically to enhance global planning performance for inland waterway networks. Whilst the algorithm developed in this study yields a performance improvement over other variations, it is questionable as to whether the minimisation of path length is the most suitable goal for maritime path planning.

The A* search algorithm was evolved for use on road networks, where the shortest path can be assumed optimal. Unlike ground vehicles however, marine vehicles must navigate through a dynamic body with varying influences and consequential resistance. The impact of which is particularly prominent on inland waterways due to inconsistent waterway depths and currents. The direction and magnitude of the current at a particular point on a waterway intuitively has an influence on the water resistance acting on a vessel. Perhaps less intuitively, the channel depth has a significant influence due to its relation with the squat effect which is caused by an increase in velocity as the distance between the hull of a vessel and the waterway bed gets smaller. The higher velocity sees a reduction in pressure, impacting on the buoyancy equilibrium, causing the sinkage of the vessel and giving the

phenomenon its 'squat' characteristic. Squatting impacts the speed and hinders the manoeuvrability of a vessel whilst also increasing the risk of groundings. A deeper waterway depth is as such favourable to reduce the impact of the squat effect and typically the deepest part of a channel is situated around centre of a waterway.

Despite the similarities in traffic behaviour and architecture, global path planning for the inland waterway cannot always be conducted in the same intuitive manner as road networks. Given the varying channel depths and currents that can be encountered, the shortest path doesn't necessarily yield an optimal route. This means that in the case of many inland vessels, the optimal route may in fact be the most fuel efficient, especially given that sustainability is a key incentive securing market share of inland freight transportation. The EconomyPlanner[22] was developed to solve this very issue on the European inland waterway network. The planner optimises by minimising the fuel consumption rather than distance, finding the best route to based upon real-time waterway depths and currents. In addition to the route, the planner also advises on optimal engine speed to ensure timely arrival at the destination and the maximum allowable loading condition to ensure safe transit given current water depths.

Despite the absence of fixed traffic lanes, inland waterways should not be considered as disorderly highways with vessels travelling in both directions competing for the path of least resistance. The results from a study into waterway capacity at the Port of Rotterdam [23] highlight that vessels do tend to follow similar, ordered paths that can be more or less divided by the two travel directions on each waterway (see Fig.2.3). The ordered behaviour can be attributed to the adherence to rule nine of the International Regulations for Preventing Collisions at Sea (COLREGs) which states that a vessel should travel on the starboard side of the waterway. Given that the water depth is typically greatest at the centre, vessels thus tend to travel starboard of the waterway centre.

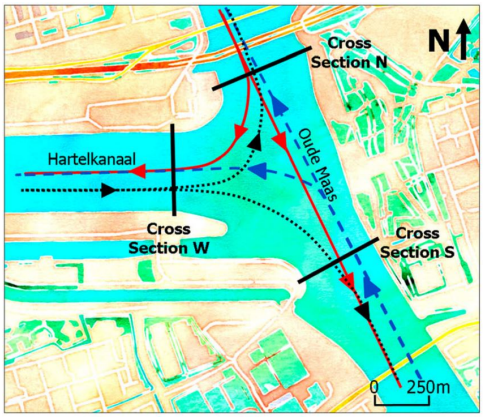


Figure 2.3: Typical Directional Paths [23]

The guidance branch of a GNC also encompasses the task of local planning. This topic has been the main focus of research into maritime autonomy to date as it involves the execution of the protocol for avoiding collisions. The typical procedure for collision avoidance is well covered in a comprehensive review of the state-of-the art by Huang et al. [24]. Unlike other reviews, this paper covers research from both manned and unmanned applications to create requisites for collision avoidance. A useful breakdown is provided of the typical sub-tasks involved, which are neatly divided into motion prediction, conflict detection and conflict resolution.

Motion prediction is fundamental to achieving collision avoidance in an environment with dynamic obstacles. Unlike with static obstacles, the motion behaviour of each target vessel needs to be assessed so that a forthcoming trajectory can be predicted and there is a number of ways that this is achieved. In some cases, simple Physics-based predictions are made by assuming the current course and speed of a contact vessel will be maintained. Although this technique offers low complexity, the straight line predictions supply unrealistic predictions outside of a close proximity range. A slightly more advanced technique is manoeuvre-based prediction which estimates a vessel's intentions based upon regulations and other decision variables. Despite the potential to create a more accurate estimation, the downside of this prediction type is that any miscalculation of intention in close quarters could lead to a collision. Arguably the most robust solution is an interaction-aware approach which relies upon cooperative data sharing between vessels, ideally of their intended paths. Whilst this approach would provide the most accurate trajectory prediction, realising this cooperation between all vessels on the waterway would carry heavy overhead burdens and likely take a long time to implement.

Conflict detection is the procedure that follows thereafter. Its role is to utilise predicted trajectories to determine whether the motion presents a risk to the autonomous vessel. The closest point of approach (CPA) is among the popular metrics used for assessment of risk [25] [26]. Risk assessment using the graphical or numerical metrics may be conducted using model-based methods with binary or probability decision making. Alternatively, experts' knowledge may be employed to configure input parameters for numerical risk analysis [24]. Whilst many solutions can be encountered, regardless of technique or metric, accurately quantifying and assessing risk is an extremely difficult task. Assessments will always remain estimations and it is arguably impossible to conduct this task without there existing an element of doubt, even for an experienced human operator.

The incorporation of radial safety regions or vessel domains during conflict detection tasks are popular in maritime solutions to autonomy [9][11][27][12][28][10][13]. Vessels travelling on the narrow channels of inland waterway networks are however frequently subject to close side-side interactions with one another. This renders the direct application of radial regions less suitable, particularly in the case of larger vessels. The elliptical regions presented in [29] or the rectangular uncertainty zones presented in [30] perhaps offer more eloquent solutions to inland vessel applications.

Conflict resolution is the final stage of local planning and the cornerstone of collision avoidance. Its responsibility lies in determining a suitable collision free path based upon all the information available from preceding tasks. Artificial potential field, collision cone and velocity obstacle approaches are some of the most frequently encountered techniques for managing local routing [31]. Yet of late, research into maritime collision avoidance has been further advancing in its maturity. More recent work includes the proposal of hybrid solutions to local planning [9] [11] [29] and emergency contingency when normal avoidance is not possible [10]. Moreover, focus has even turned away from the generation of merely feasible avoidance and towards optimised avoidance by using a rolling horizon technique to select optimal heading angles [32]. Advancements in artificial intelligence have also lead to their consideration in solving the problem of maritime collision avoidance [33][34][26].

The importance of COLREG integration within the conflict resolution stage is unquestionable. For autonomous vessels to be realised, the navigation system must be compliant with these rules. Multiple techniques have been proposed to integrate COLREG compliance into the local planner. Fuzzy rules have been integrated into a modified Virtual Force Field approach [27] which showed promising results during simulation, but the complexity of the system and unwavering adherence to the regulations make the technique less appealing. Benjamin et al. [35] recognise that compliance is not a binary problem and the interval programming approach allows for decisions to be made by solving an optimisation problem with weighted functions. This technique has further matured and is now implemented as a module [25] in the open source MOOS-IvP platform [36], a platform which has been utilised in many applications, including the inland waterway [7].

As with global planning, many of the solutions for maritime collision avoidance have been adopted from other sectors. In some cases, these solutions have seen adaption and evolution prior to implementation, in other cases they have been more directly implemented. For example, the open source local path planning technique of [37] designed for ground vehicles has been utilised for inland surface vehicles on the canals of Amsterdam [8].

2.4. Collision Regulations

The navigation rules on waterways worldwide are governed by the International Regulations for Preventing Collisions at Sea (COLREGs) [38]. Local exemptions to the rules are rare and only put forward in exceptional circumstances so to avoid confusion by maintaining the global standard. A total of thirty-eight rules are present within the regulations, however only those deemed of main importance for the developmental application will be further divulged hereon. The rules will be detailed textually and the number of the rule shall follow in brackets. It is noteworthy to mention that this summary only considers interactions between powered vessels and neglects the special rules regarding interactions with sailing boats. Figure.2.4 provides a graphical aid to the text and includes the key interactions.

The preliminary attention of the regulations is focused on detailing the applicability of the rules (rule 1) and the responsibility of all parties(rule 2), along with the main definitions(rule 3). To summarise the application of the forthcoming, the rules apply to all vessels navigating on waters navigable by seagoing vessels, however on some inland waterways special rules may apply for which the skipper should pay attention. Special rules will however rarely contradict or interfere with this protocol. On the topic of responsibility, it is stated that the rules do not exonerate any party from the consequences entailed when neglecting to consider exceptional circumstances during the adhesion to this standards.

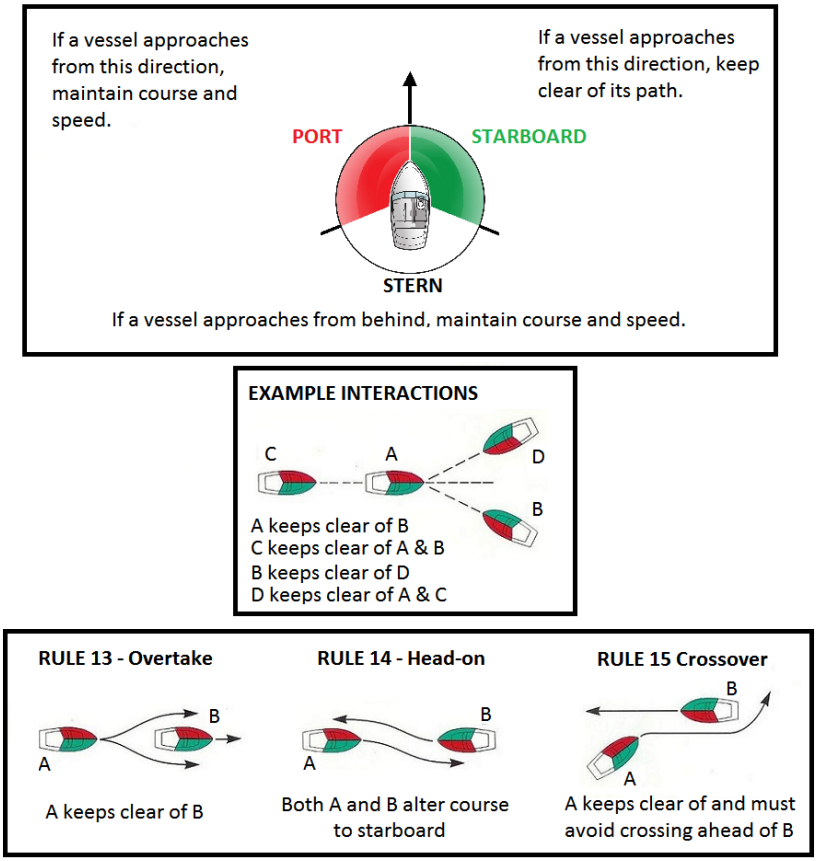


Figure 2.4: Applicable Collision Regulations.

In any condition of visibility (rule 4), a vessel should have a consistent means of perception so that a full appraisal of the current situation can be made (rule 5). Necessary equipment should be present, in working condition and used correctly so to ensure that the vessel is in a position to best avoid a collision given the prevailing circumstances (rule 6). The speed at which the vessel proceeds should also at all times be appropriate to these prevailing conditions, so that she can avoid collisions or stop within a reasonable distance (rule 7).

To avoid a collision, action should be taken in good time according to the rules. If the scenario permits, the action taken should be significant enough to be noticeable to other vessels, small successions of alterations should be avoided. If possible, course should be altered to avoid close quarter encounters and when these encounters are unavoidable a safe distance should be exercised until clear. If a clear decision cannot be made immediately, the vessel speed should be reduced until greater clarity is attained and an informed decision can be made (rule 8). In narrow waterways, a vessel should travel as near to the outer limit of starboard side channel as is safe and practicable (rule 9). A vessel should adhere to traffic separation schemes (TSS) as applicable. The vessel should remain in her lane, follow the general direction of traffic flow and stay clear of the separation line as far as possible. When joining a traffic lane, the vessel should do so at an angle as close to parallel as practical (rule 10).

The action taken in visibility conditions where two vessels are in clear sight of one another (rule 11) are considered for this project, other scenarios will be neglected at this stage of development. A vessel conducting an overtake manoeuvre must keep out of the way of the vessel being overtaken, which may maintain its course (rule 13). Two vessels which are approaching one another head-on shall each shall alter her course to starboard so that they pass port-to-port (rule 14). In inland waterways of course, a vessel is limited to furthest extremity of the navigable channel on its starboard side and so the respective traffic separation scheme should be referred to by both parties when deciding on action.

In a cross-over scenario, the vessel which has the other on her starboard side must keep out of the way and should avoid crossing the front of the vessel. Instead taking action to navigate behind the other vessel if required (rule 15). For clarity, this rule does not hold for overtaking vessel, as even in the case of bearing changes, the vessel overtaking remains responsible for clearing the vessel that is being overtaken. I.e. if being overtaken from her starboard side, a vessel is not expected to take action based upon rule 15, the overtaking vessel has a duty to achieve clearance during the manoeuvre.

The give-way vessel is required to, so far as possible, take timely and substantial action to keep well clear of the stand-on vessel (rule 16). The specific action of the give-way vessel should be in accordance with the rules of this set of regulations. The stand-on vessel is required to maintain her speed and course during an encounter, so to allow the give-way vessel to take predictable action (rule 17). If it becomes apparent to the stand-on vessel that the give-way vessel has not acted appropriately to avoid collision, she shall take such action as will best aid collision-prevention. Any action taken by the stand-on vessel to this end, does not however relieve the give-way vessel of her duty in the encounter. Except where rules 9, 10 and 13 contradict, a vessel should always keep out of the way of any vessel not under command, e.g. a moored vessel (rule 18).

2.5. Performance Evaluation

With stereovision taking a pivotal role within this research focus, it is important to understand the quality of the Navigation data that can be provided by the sensor technology. This assessment can best be made through applying key performance indicators to the perception and localisation systems themselves. Evaluation metrics deemed appropriate for these tasks are introduced in subsections 2.5.1 and 2.5.2. Evaluation metrics are also selected and proposed in subsection 2.5.3 for the assessment of collision avoidance performance during experimental testing.

2.5.1. Localisation System

From the pose data obtained from a localisation sensor, there exist two main data sets of interest. Primarily that of position, which for a surface vessel is within a two dimensional reference frame. Secondly, the yaw rotation angle is of interest as it is directly used to calculate the heading of the vessel.

As the localisation data has a direct influence upon the collision avoidance behaviour, sufficient data is critical. Consistency in measurement is arguably more important than high accuracy as knowing location to within a millimetre will have little impact upon avoidance performance, whereas large errors in measurement, regardless of how infrequent could have critical consequences. Therefore rather than utilising a simple metric taking the mean deviation error, the root mean square error shall instead be taken. This metric has a sensitivity to outlying measurements [39] therefore highlighting the occurrence of large errors which would be detrimental to collision avoidance performance.

The procedure of Zhang et al. [40] presents a particularly well-suited method for acquiring evaluation metrics from Visual-SLAM approaches, with the absolute and relative trajectory errors providing apt means for performance evaluation. The relative position and rotation errors can be evaluated in box plot form with measures covering multiple trajectory distances to highlight the impact of localisation drift. The absolute trajectory error (ATE) describes the aforementioned root mean square position and rotation error and can be used to quantify the quality of the whole trajectory estimation. The definitions the two ATE metrics can be defined below in Eq. 2.1 and 2.2 relating to the position and rotation error respectively with $\hat{p}_i$ and $\hat{y}_i$ being the ground truth values, p_i and y_i being the measurement values and N being the number of samples.

$$ATE_{pos} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{p}_i - p_i)^2}{N}} \tag{2.1}$$

$$ATE_{rot} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{N}} \tag{2.2}$$

2.5.2. Perception System

The two main tasks of the perception system is to detect obstacles and determine their position. The position of an obstacle determined by a navigation system can be evaluated by comparing the obtained result to that of the

accepted ground truth. The deviation of the attained position and the ground truth value provides an indication of accuracy and can quantified by calculating the root mean square error from a range of measurements in the same way as the localisation absolute trajectory error. The definition of obstacle position error is given in Eq. 2.3 below with $\hat{o}_i$ referring to the ground truth position of the obstacle, o_i being the measured position value and N being the number of samples.

$$RMSE_{pos} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{o}_i - o_i)^2}{N}} \tag{2.3}$$

One of the most prevalent metrics for evaluating detection performance is the relation between precision and recall which can be illustrated by a graphical curve [41]. If the precision of an object detector remains high whilst the recall increases, performance can be considered to be high. Precision and recall are defined as shown in Equations 2.5 and 2.6 respectively. The IOU threshold stated in the definitions is used to limit the acceptable precision of a detection and is quantified by means of a ratio using the relation in Eq. 2.4. IOU stands for Intersection over union and measures the overlap that exists between the object bounds detected and the ground truth bounds as illustrated in Fig. 2.5. Three standard thresholds are used, 0.5, 0.75 and 0.95, which are listed in ascending order of accuracy.

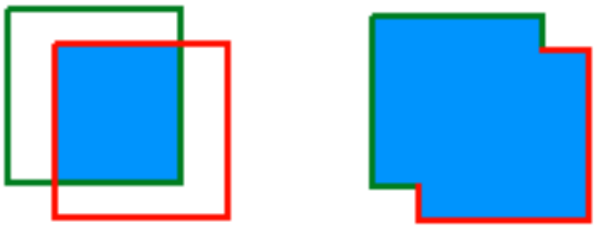
$$IOU = \frac{area\ of\ overlap}{area\ of\ union} \tag{2.4}$$

$$Precision = \frac{correct\ predictions}{all\ predictions} = \frac{TP}{TP+FP} \tag{2.5}$$

$$Recall = \frac{correct\ detections}{all\ ground\ truths} = \frac{TP}{TP+FN} \tag{2.6}$$

$where$:

- TP = True Positive: A correct detection above IOU threshold.
- FP = False Positive: An incorrect detection outside of IOU threshold.
- FN = False Negative: A missed detection.



(a) Intersection region (b) Union region

Figure 2.5: Illustration of Intersection over Union [41]. *Green box represents ground truth, red box represents detection result.*

2.5.3. Collision Avoidance

Applicable key performance indicators for collision avoidance were selected by referring to the paper of Nous et al. [42]. As the scope of testing in this project will be limited to collision avoidance with a single contact vessel and a consistent environment, only two of the performance metrics from the paper will be applied in this research project. Furthermore the exact definitions of the avoidance metrics are adapted to better meet the specific maritime application as opposed to the original UAV application.

The two performance metrics that will be used to evaluate the performance of the collision avoidance system are success rate and path optimality. The success rate provides an indication of a systems fundamental capability to avoid collision. Whereas the path optimality reviews how close the path selected by the avoidance system was

to a most desirable selection. A hierarchy exists between the former and latter metric as the path optimality is dependent upon a successful avoidance case for the assignment of an optimality value.

The success rate in this application shall be divided by three outcomes, in the same way as the reference with each carrying a percentage value relating to occurrence. Rather than following the direct success definitions however, this paper shall apply a new interpretation based upon the integration of maritime collision regulations in the success evaluation. Consequently the outcomes are selected to be: compliant collision avoidance (compliant that is with expected protocol), semi-compliant collision avoidance (collision avoided but procedure not adhered to exactly as expected) and finally failed avoidance that would lead to a collision.

Path optimality will be divided into two secondary measures and manipulated to form a single metric. One concerning the additional distance the local avoidance path has added to the global route and the other focusing on the collision risk that the path resulted in, quantified by the shortest distance between the autonomous vessel and obstacle during avoidance. As these two optimality metrics are correlated to one another, a simple optimality factor can be created as a metric to quantify this relation and define an overall optimality. The definition for said metric can be found in Eq. 2.7, where the numerator refers to the added distance that the avoidance path has added and the denominator states the closest point of encounter between the vessel and the obstacle during avoidance. A lower metric value indicates optimality.

$$Path\ Optimality = \frac{Additional\ Path\ Distance\ (m)}{Collision\ Vicinity\ (m)} \tag{2.7}$$

2.6. Conclusion

This Chapter has addressed the research question *What is the state-of-the-art in inland autonomy and collision avoidance research?*. Maritime autonomy has been widely researched however a focus on inland applications and the specific challenges the environment presents has only recently emerged. It was found that a number of sensors are typically utilised onboard autonomous vessels, particularly inland vessels which require due to closer interactions require low-mid range coverage. Whilst stereovision is utilised, the sensor takes a back seat role with LiDAR still being the preferable choice. It was however discovered that autonomous cars of late favour vision based technology as a primary source of sensor data, using multi-device setups to achieve sufficient coverage and dropping the dependency on LiDAR. A gap was identified in inland autonomous research to this regard, with the possibility that vision sensors could provide a lucrative research direction to the field in the same way they have assisted in the evolution of autonomous ground vehicles.

A number of guidance approaches were identified in the literature for autonomous vessels, with open water applications being well covered. The restricted channels synonymous with the inland waterway however do not allow for the direct application of existing techniques for conflict detection and resolution. A further gap was highlighted here, whereby further research was required to tailor a collision avoidance procedure to the inland waterway environment. The importance of experimental testing was highlighted in the review and a number of suitable key-performance-indicators were further selected to assist in the evaluation of navigation and guidance system performance. The most applicable evaluation metrics were selected for stereovision based perception and localisation and metrics were identified for quantifying the collision avoidance performance.

3

The Autonomous Vessel

The realisation of inland autonomy requires a vessel with a confluent hardware and software structure. The Guidance, Navigation and Control System structure that was introduced in the previous chapter sees continued use throughout this Chapter and thesis. This Chapter is specifically dedicated to providing an overview of the autonomous vessel system, introducing both the hardware components utilised and the approach to software implementation. The latter attention to software implementation also answers the fourth research question: *How can the guidance and navigation modules be implemented towards software standardisation and accessibility?* and is covered in Section 3.2.

3.1. System Hardware

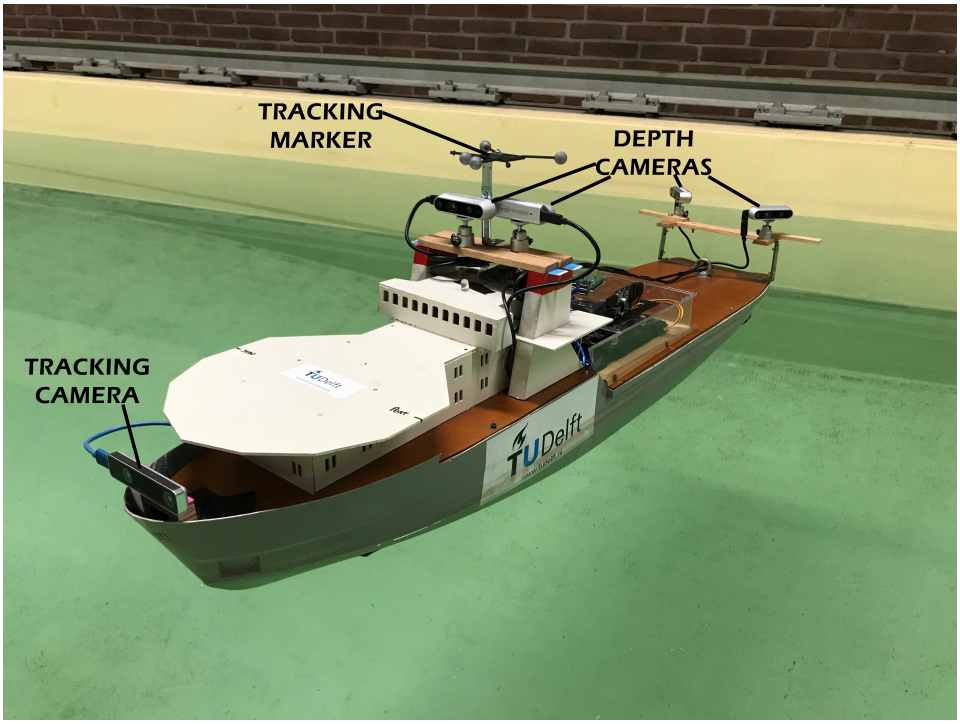
The developments made in this project shall be applied and tested on a model scale test vessel provided by the Researchlab Autonomous Shipping. The 'Grey Seabax' vessel in question can be seen below in Fig. 3.1a. Albeit a model of a seagoing vessel, the length to beam ratio of the vessel can be comparable with a number of commercial inland vessels and subsequently deemed suitable for this level of research. The vessel has a length of approximately 1.4m and a beam of 0.3m and is actuated by means of four azimuth thrusters located at the bow and stern of the craft. Actuation is managed by an Arduino control unit onboard the vessel that communicates with the high-level control system running aboard a host PC via the ROS network, using a Raspberry Pi 3b as a communications bridge.

As the new Navigation System shall utilise stereovision as its primary sensor set, as well as the addition of new sensors to the system hardware, an onboard processing unit is also required to acquire data and handle on board processing. To enable the processor and sensors to operated, a number of ancillary devices were also required to handle power and data supplies. A full list of the hardware editions to the vessel can be found below, with further discussion thereafter.

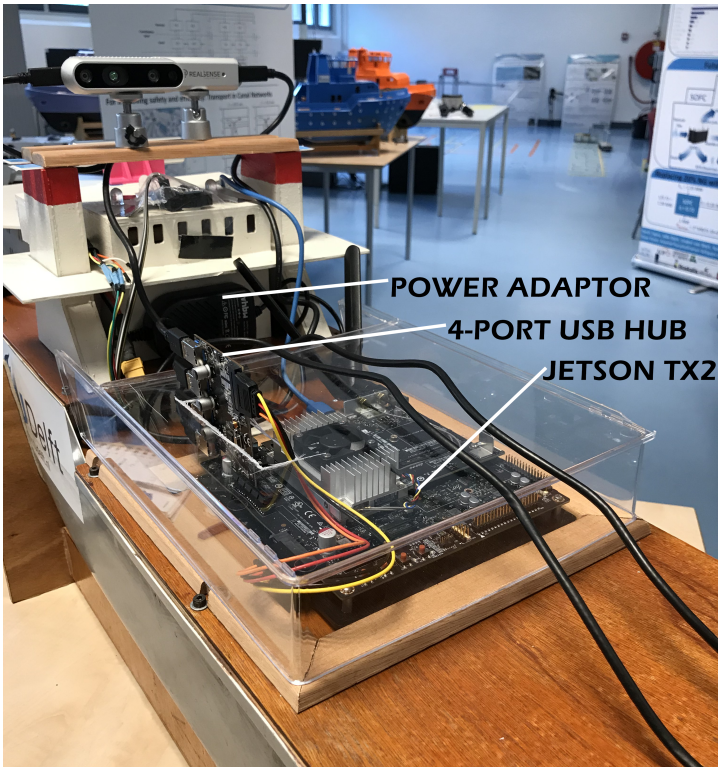
- NVIDIA Jetson TX2 Development Kit (stock)
- Power Adaptor - On-board battery to TX2 - 12V to 19V, 4.74A (90W)
- PCIe Expansion Card - 4 ports USB3.0 - SATA power
- Four Intel RealSense D435i Depth Camera
- One Intel RealSense T265 Tracking Camera

The sensors selected for gathering environmental information are all part of the Intel RealSense stereovision range. Each of these stereovision devices encompasses multiple visual sensors and an inertial measurement unit. The devices are provide a near out-of-the-box solution to stereovision by using built in processing units to deliver processed data directly to the host. For perception tasks, the D435i devices were selected to provide high level depth accuracy along with a supplementary colour stream. For localisation, the T265 tracking camera has been selected for its optimality at performing localisation tasks. The depth device executes stereo algorithms on board, allowing the disparity frame to be directly accessed from the device. The tracking camera, performs Visual-SLAM on board and provides a direct stream of pose data to the host. The selection of devices which

perform these primary processing tasks on board provides a significant head-start to such a navigation system and enables the focus to be transferred to more critical secondary processing challenges.



(a) The Vessel, its Onboard Sensors and Passive Tracking Marker



(b) On-board Processing Unit and Ancillaries

Figure 3.1: The Grey Seabax Autonomous Vessel

An onboard processing unit is required to receive and process all the sensor data and the device selected for this task is the NVIDIA Jetson TX2 Development Kit. The main reason for its selection come as a consequence of its small form factor and high graphical processing power, ideal for artificial intelligence tasks. All the stereo devices need to be connected to the Jetson via USB bus, yet only one port is provided as stock, meaning an extra four ports are required. As the RealSense devices are relatively power and data intensive, the additional ports must also be capable of providing USB3.0 standard power and data transfer rates. To achieve this a PCI-Express card has been added, drawing power from the SATA power supply. For the Jetson to operate at full computational capacity, it requires a power supply of 19v and 4.74A usually supplied by a mains adaptor. The Seabax vessel which the Jetson will be mounted upon however has an onboard battery which supplies 12V. To overcome this power supply issue, a power adaptor is used to step up the voltage and manage the current to supply a max output of 90W, thus allowing the board to run without operational constraint. An overview of the processing setup on board the vessel can be seen in Fig. 3.1b.

As well as an onboard processing unit, a host PC is also required for the vessel to be operational. Although an entirely decentralised setup is feasible, with this stage of development and this scale of vessel it is most logical to maintain the host PC as the core unit. The host PC used for this project is a MacBook Pro A1278, which has a 2.5GHz Intel Core i5 processor and 8GB of 1600MHz RAM. The specific roles of the on-board processor and the host processor shall be discussed within the following section.

3.2. Software Structure

The fourth research question regarding the implementation approach towards software standardisation is addressed within this section. Subsection 3.2.1 reviews the potential implementation approaches suitable for the application and which selections are made for existing autonomous software platforms. Subsection 3.2.2 then discusses the selected implementation approach for this application with the remaining subsections detailing the individual sub-system structures.

3.2.1. Background Research

The procedure for implementation of an autonomous vehicle system requires two fundamental decisions to be made. Firstly, regarding the selection of a suitable middleware and secondarily the selection of a primary programming language. During the development of open-source software is it is important that one considers future accessibility to the wider community as opposed to purely the preference of the researcher. The structure that is then built around this foundation should ideally be configured in a well-ordered modular manner so to expand potential use cases and make the system adaptive to ongoing developments.

Regarding the selection of middleware, two main contenders present themselves for this use case within the field of autonomous shipping, MOOS and ROS. The Mission Oriented Operating Suite (MOOS) was developed at the University of Oxford by the Mobile Robotics Group and integrates a publish–subscribe style framework as well as applications specific to both land and marine robotics applications. The MOOS-IvP application for collision avoidance that was introduced earlier in this chapter makes use of this middleware framework. The MOOS-IvP is a collection of open source modules developed to provide autonomy for robotic platforms, in particular unmanned marine vehicles. These modules form a MOOS Application which can be run within a MOOS Community to make use of a communications protocol with other application processes within the community.

The Robot Operating System (ROS) provides another middleware framework, again offering a standardised communications protocol as well as additional tools and capabilities making it a very popular choice throughout all branches of the robotics field. As well as a robust communications network, there are a host of additional tools for the system developer, not to mention a large open source community where custom packages are made available. Furthermore, ROS offers good support to both Python and C++ programming languages through specialist packages. ROS is for example the chosen middleware of the Autoware project for autonomous ground vehicles who offer a modular structure of ROS packages with the language of choice again being C++.

Despite the two example cases choosing to utilise C++ as the programming language of choice for their autonomous application frameworks, it is recognised that other programming contenders exist, even if they do not offer the same execution speeds. When considering the best programming language to use for the implementation of an autonomous vessel applications, it can be best to consider the accessibility to ongoing research. The more people who can access and fully utilise an open source project, the greater the effective contribution could be to the field in the long term.

A study of over one hundred thousand open-source projects found Python to be the second most popular main programming language overall [43]. Furthermore when considering the background of researchers working within the field of autonomous shipping, the majority belong to maritime/mechanical engineering departments. Within the mechanical field, degree curriculum's have recognised the crucial role computing and programming skills serve. Particularly since the advent of Industry 4.0, the integration of Python as well as MATLAB courses within the education programs has become the norm[44]. Even within the field of computer science, Python appears to be the primary choice for inclusion within education programs [45].

Python is the frequently the top choice for data analysis and processing tasks. Machine learning, artificial intelligence and deep learning tasks are far more accessible due to the wealth of libraries that are available to the user. These technical domains present a plethora of opportunity within the field of autonomy and it can be expected that maritime research within this area shall continue to move in this direction, with advancements already being underway. Furthermore, from a broader perspective, Python has a reputation for being particularly user friendly, with its simple syntax making it very readable and the approach of object orientated programming being flexible and intuitive. The interpreter framework does offer the user simplicity over the use of compilers however does so with a reduction in execution speed. Nonetheless, for particularly demanding tasks a superset of Python could be used to benefit from compiled speeds from within the Python environment [45].

3.2.2. Selected Approach

The devised software structure is introduced hereon based upon conclusions drawn from the previous subsection and the challenges of the specific application. The favoured Guidance, Navigation and Control structure also forms the framework of the software implementation, with the system applications being built underneath each of the main topics. A total system overview can be seen in Fig. 3.2 and the rest of this subsection is dedicated to detailing the chosen implementation approach.

Due to the plethora of existing tools and capabilities, the Robot Operating System has been selected as the ideal middleware for this system and future developments of it. The NVIDIA Jetson TX2 on-board the vessel is flashed with Jetpack 4.4 and runs on the stock install of Ubuntu 18.04 with an install of ROS Melodic. The host PC runs on Ubuntu 16.04 and and install of ROS Kinetic. The ROS1 framework has been utilised at this stage as it carries Long Term Support and provides the best compatibility with the majority of tools and packages. It is however noted for future versions that a migration to ROS2 would be beneficial, not least due to the fact that a number of the packages require the use of Python 3 for operation which is not officially supported by ROS1.

The ROS network allows for communications links to be setup between various sub-systems and devices through a publisher/subscriber protocol. The host PC acts as the ROS Master during operations, with the other boards acting as slaves within the network. As seen in Fig. 3.2, the Guidance and Navigation Systems have ROS packages to handle the major tasks that benefit from being kept independent of one another. This represents a key modular aspect of this implementation whereby one package within the system can be utilised without the full GNC having to be online, expanding the possibilities of future application within other open-source projects.

It can be noted that the Control System is not implemented within a ROS package as this sub-system is not a research focus of this project. The control system does nonetheless utilise the ROS network to handle communications from the Guidance and Navigation Systems whereby it receives setpoints. Further communication is handled by ROS between the control system on the host PC and the onboard low level control system. The control system and its composition shall be further discussed in the third sub-section.

The implementation of the Guidance and Navigation systems in this project are written using Python and as briefly mentioned, embedded within ROS packages, four to be precise. To enable future research developments within the ResearchLab Autonomous Shipping and the wider research community, it was concluded from the literature review that Python offered the optimal programming language for the use case. It can further be observed that the software modularity continues even at a sub-package level, again further promoting future research development by allowing focus on the optimality of specific tasks as opposed to the development of an entire system. The packages themselves shall be further discussed in the next two sub-sections.

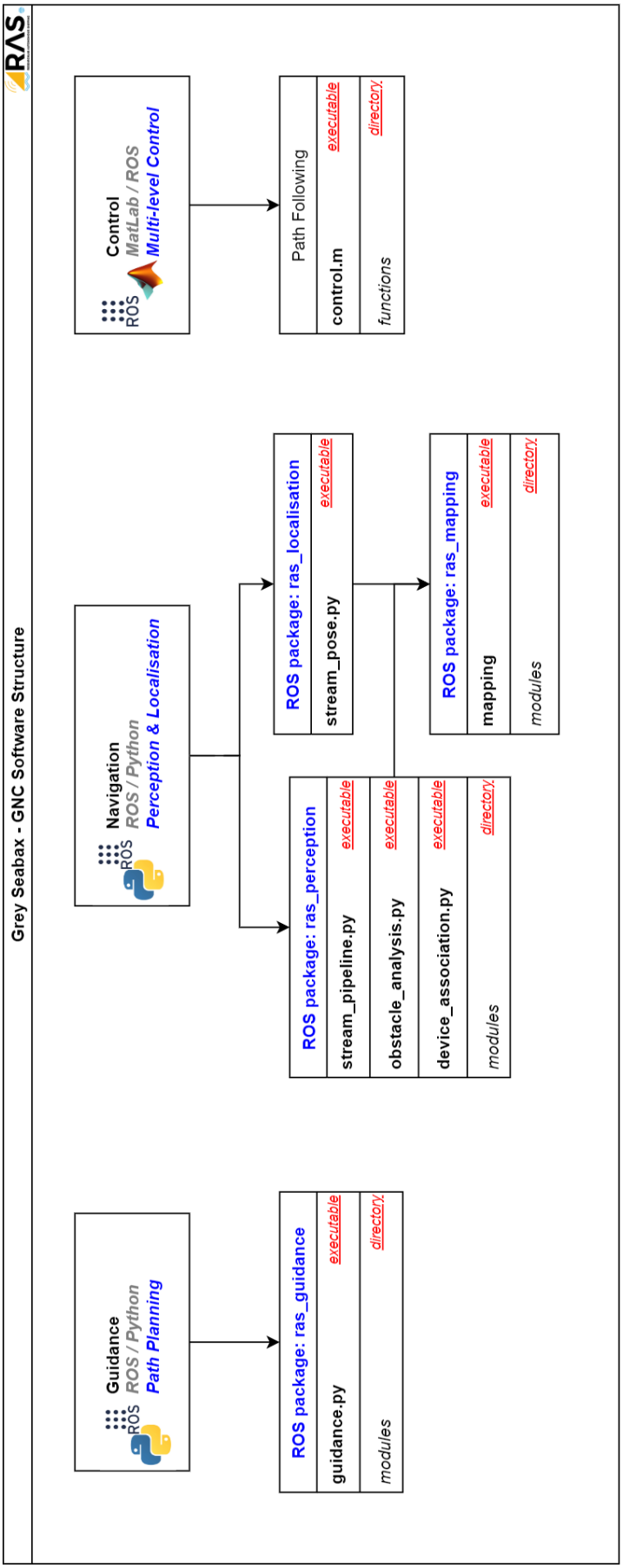


Figure 3.2: Overview of GNC Software Structure

3.2.3. Navigation System Software

An overview of the software structure seen in Fig. 3.2, illustrates the Navigation System and its three main branches for the ROS packages covering the tasks of perception, localisation and mapping. Within the packages are executable Python scripts which are used to conduct specific sub-tasks. The procedure of individual sub-tasks within the navigation system will be detailed within the following. The remainder of this subsection shall provide an overview of the software structure, all of which is implemented in Python and utilises the ROS middleware. A detailed explanation of the executables and module roles shall be provided in Chapter 4.

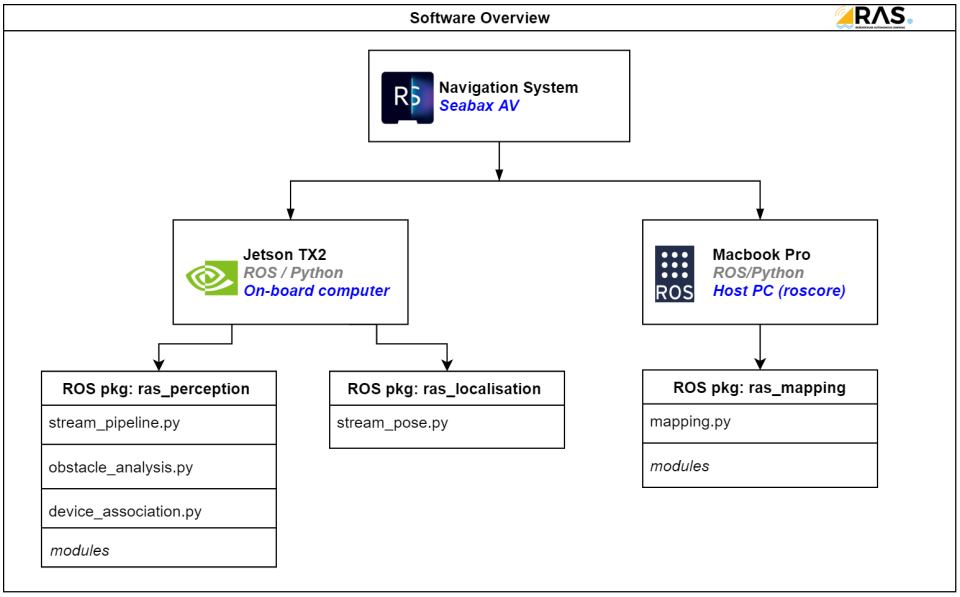


Figure 3.3: Navigation System Software Setup

The navigation system software contains a total of three ROS packages together covering the major tasks of perception, localisation and mapping. The perception and localisation systems are responsible for gathering and processing of sensor data from the cameras and remain independent as they have potential applications independently to one another. The mapping system is in charge of translating the data from these two packages to locate both the autonomous vessel and any obstacles into local and global frames before determining the derived attributes and streaming this data as the Navigation System output. Fig. 3.3 provides a secondary breakdown of the Navigation System structure detailing the processing unit where each package is stored and run on.

The perception and localisation packages could be directly applied to any other system utilising the same sensors, meaning that other vessels can be easily equipped with similar stereovision setups and achieve plug and play perception and localisation, even with a different number of depth sensors. The mapping package is kept separate as it may or may not be required by other applications and in the case that it were, application-specific adjustments to the constants based upon the positioning of the sensor(s) on the vessel and the global environment of operation would need to be made.

Module Name	Dependant Program	Function
DeviceConfiguration	stream_pipeline device_ association	Configuration of the depth devices and management of data streams.
ObstacleDetection	obstacle_analysis	Handling of obstacle detection on all four device streams.
ObstacleLocalisation	obstacle_analysis	Determination of obstacle position through manipulation of the depth frame data.
ObstaclesClass	obstacle_analysis	Responsible for the creation and management of obstacle instances throughout obstacle analysis.

Table 3.1: Perception Package - Modules

The localisation package is significantly less complex than the perception package with only one executable being present which enables the sensor, processes the data and publishes pose data via ROS. The perception system is initialised using two ROS executables, namely 'stream_pipeline' and 'obstacle_analysis'. The former is responsible for enabling the four depth cameras, logging device intrinsic data and streaming frame data via the pipeline node. The obstacle analysis node is then responsible for detecting obstacles within the frames and consequently deriving the relative obstacle position. The modules within the package sub-directory enable these individual tasks. These modules within the perception package cover a number of tasks as summarised below in Table. 3.1.

The multi-device setup introduced in the previous subsection makes use of four depth cameras to perceive the area around the vessel, with one device assigned to cover the fore, aft, port and starboard sides respectively. When first mounting the depth cameras, or after any adjustments to the hardware configuration, the individual devices must have their serial number associated with their position on the vessel. Device association can be achieved by running the dedicated executable (device_association) within the ROS package and following the on-screen queues when prompted by the GUI. After all the devices have been associated to their respective locations, the configuration is saved to a file stored within the package's modules directory. After this configuration is set the main perception program automatically re-associates the devices upon execution.

Module Name	**Dependant Program**	**Function**
ObstacleTracker	mapping	Associates obstacles between iterations, assigning them with an ID number
MappingClass	mapping	Responsible for the creation and management of AV and obstacle objects and their attributes throughout the mapping procedure
AVDomain	mapping	Maps obstacles inside the AV domain reference frame from individual sensor feeds
GlobalFrame	mapping	Maps the AV and obstacles within the defined global coordinate system
Plotter	mapping	Generates a pseudo-radar plot with any obstacles detected in the environment displayed along with their ID number.

Table 3.2: Mapping Package - Modules

The mapping package only contains one executable that is responsible for generating the Navigation System output from the available localisation and perception data. To achieve this a number of sub-tasks are required in order to conduct translations into the local and global frames and these are conducted by the callable modules within the packages sub-directory. A summary of the mapping modules is provided in Table. 3.2.

3.2.4. The Guidance System

Reviewing again the software structure seen in Fig. 3.2, it can be seen that the Guidance System only consists of one ROS package, with one executable. Running the guidance executable initiates both the global planner and local planner which are stored within the modules directory. The global planner only contains one module that imports the global path that has been provided and creates a path object with waypoints for the guidance system. The local planner on the other hand contains a number of modules that are configured to handle collision avoidance procedure and are detailed in Table. 3.3. A detailed explanation of the executable and local planner module roles shall be provided in Chapter 5.

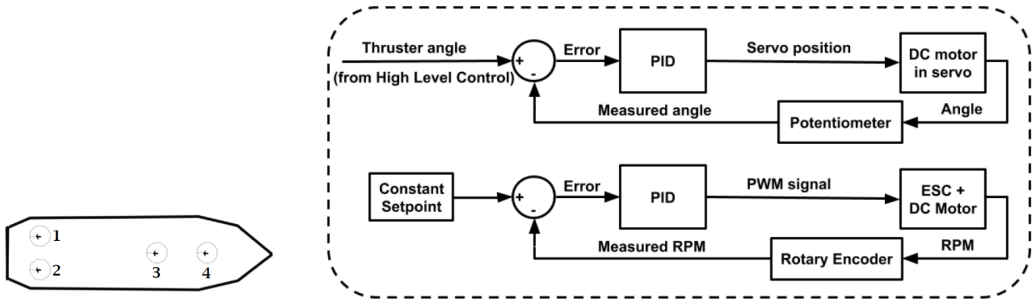
Module Name	Dependant Program	Function
MotionPrediction	guidance	Handles the forecasting of contact vessel trajectory.
ConflictDetection	guidance	Responsible for monitoring conflict with contact vessels during interactions and triggering a response where necessary.
ConflictResolution	guidance	Determines the required action that should be taken in each case of conflict.
GuidanceClass	guidance	Manages the Autonomous Vessel, Contact Vessel and Path objects throughout the guidance system

Table 3.3: Guidance Package - Local Planner Modules

3.2.5. The Control System

Although the control system does not compose part of the research focus of this paper in order to provide total context to the reader and complete the GNC overview, the control system structure shall be presented within this subsection. From a broad perspective, the main control loop is built within a MatLab environment that is run on the host PC and communicates with the marine craft via the ROS network.

The existing control system for the Grey Seabax test vessel prior to this project already had a multi-level control structure to handle a simple path following control strategy using sensor data from a GPS and IMU sensor to complete the feedback loop. As this project replaces the sensor data input with that produced by the new navigation system and requires waypoints to be set by the new guidance system, the high level control structure required some alterations.



(a) Azimuth Thruster Locations (b) Control Structure

Figure 3.4: Grey Seabax Low Level Control

The Seabax vessel has total of four azimuth thrusters and their locations on the ships hull can be seen in Fig 3.4a. Two stern mounted thrusters (1 & 2) and two bow thrusters (3 & 4) can provide actuation in the three degrees of freedom of surge, sway and yaw. The path following technique utilised however only controls actuation in yaw from the heading error, with the the engine speed not being regulated to control surge velocity and sway not being simultaneously controlled. The control of vessel yaw moment requires varied actuation of the azimuth thruster based upon its location on the ship, thus requiring the high level controller to further specify the individual target thruster angle for each thruster. Low level control then actuates the target azimuth angle of each of the vessel's thrusters in response to the provided setpoints, with speed being controlled by a constant setpoint as illustrated in Fig. 3.4b.

This new high level controller structure can be seen below in Fig. 3.5. The target waypoint, position and heading inputs are all received via subscription to the respective ROS topics published by the navigation and guidance systems. The target waypoint and current position of the vessel are used to generate a reference heading that is consequently compared to the current vessel heading and the relative error is defined. This heading error is then fed through a PI controller, with the proportional action (P) being used to counter-act the heading error and the integral action being utilised to attenuate disturbances. A derivative action was omitted from this controller as the dampening effect it provides proved inessential.

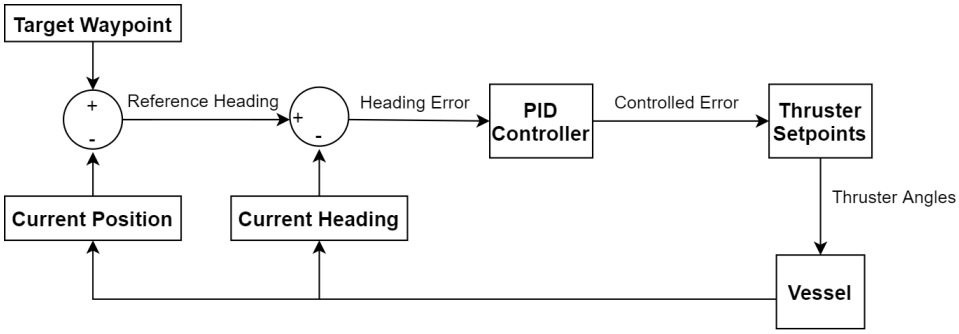


Figure 3.5: Grey Seabax High Level Control

3.3. Conclusion

The autonomous vessel hardware and software structure was introduced in this Chapter. A system overview provided context for the two consecutive Chapters that will cover the detailed approaches to navigation and guidance. The selected approach for software implementation was addressed, with which answering the fourth research question *How can the guidance and navigation modules be implemented towards software standardisation and accessibility?*.

Incentive for the chosen implementation approach was to move towards an open source platform for autonomous inland vessels. It was concluded from the research that the Robot Operating System would provide the best middleware framework for the implementation of the Guidance and Navigation units. To enable future research developments within the ResearchLab Autonomous Shipping and the wider research community, it was also concluded from the research that Python offered the most suitable programming language for the use case.

The implementation saw the use of ROS packages for the grouping of major tasks and the approach to software modularity continues even at a sub-package level. This approach was selected to promote future research development by allowing focus on the optimality of specific module tasks without the need to develop or revise an entire system. A total of four packages were developed, with one dedicated to guidance tasks and three being dedicated to the stereovision based navigation system. The separation of the navigation tasks allow for the modular application of these tasks independently of one another in other application cases.

4

Stereovision Based Navigation System

This chapter explores the implementation of a navigation system whose primary sensor set relies on stereovision for mid range perception and localisation tasks. Through which, the second research question of this paper shall be answered, specifically: *How can perception and localisation tasks be achieved by a navigation system using stereovision?* with the sub-questions below breaking down each task.

2a *How can stereovision provide mid-range perception onboard a vessel? (Section 4.2)*

2b *How can stereovision sensors be applied to accomplish localisation? (Section 4.3)*

4.1. System Overview

The role of the Navigation System is to acquire sensor data from the vessel and manipulate this data to yield useful information about the autonomous vessel and its environment. The proposed GNC system structure requires the Navigation System to provide the Control System and Guidance System with the relevant position and orientation data of the Autonomous Vessel and the Guidance System further requires information on the obstacles in its surroundings, in this case specifically contact vessels. This section will introduce the proposed structure for a stereovision based Navigation System, providing both an introduction to the system's procedural approach as well as the sensor hardware that has been selected and its configuration to suit the inland waterway.

4.1.1. Procedural Overview

In the previous chapter, three branches of the Navigation System were introduced, namely perception, localisation and mapping. In an autonomous vehicle, the navigation system must assume the roles of perception and localisation in the absence of a human controller. Perception and localisation are achieved by a human operator simultaneously, through the constant visual scanning of the surrounding environment in conjunction with the monitoring of nautical instruments. Whilst being habitual in nature to a human operator, these tasks present a significant challenge for an autonomous navigation system. Gathering visual environmental data comparable to that of human sight is achievable using modern stereovision sensors, however the processing of this raw data into usable information presents a major challenge.

Perception and localisation have already been discussed due to their focus being at a forefront of this research project, however there is an additional procedure that is required to make process of the data acquired by these sub-tasks. Mapping describes the translation of perception and localisation data into specific reference frames, be it a global or local reference frame. The result of this final processing step provides the final outputs of the Navigation System and the entire procedural flow can be visualised with the overview illustrated in Fig. 4.1. Sections 4.2, 4.3, 4.4 of this Chapter will detail the content of each sub-system.

4.1.2. Sensor Selection and Configuration

A fully operational navigation system of a vessel must have the capacity to perform perception and localisation under any condition of visibility to comply with collision regulations. Focus shall be narrowed here and the performance of stereovision shall be reviewed in clear, well lit conditions. It is recognised that for sailing during low

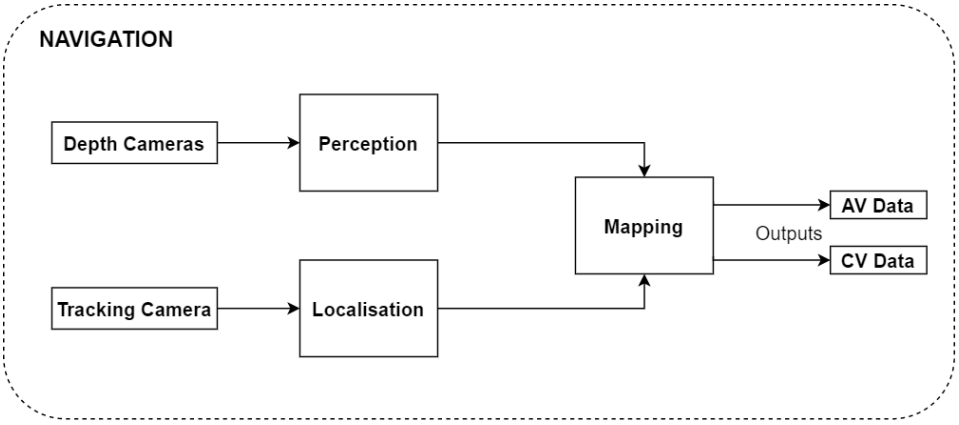


Figure 4.1: Navigation System Overview

light conditions and/or poor weather conditions, that the developed system would require at least one secondary sensor set to supply sufficient data. Furthermore, the navigation system of an autonomous vessel will likely make use of sensor fusion to improve accuracy and reliability in all sailing conditions and varying circumstances. A thorough understanding of the advantages and limitations of each individual sensor set is consequently imperative for optimal equipment selection and the fusing of sensor set data.

Regarding perception, rule five of the collision regulations states that a vessel must maintain a proper look-out at all times so to make a full appraisal of the situation and the risk of collision. Whilst a human operator will never be able to maintain a look-out in all directions surrounding the vessel, at all times, their vision does not have a fixed orientation, providing flexibility in field of view. When perceiving the environment using stereovision devices, such flexibility is not achievable in the absence of a rotating mount, such as those used by LiDAR and Radar sensors. Nonetheless, the affordability of stereovision technology enables the use of a multi-device setup, which could arguably achieve a more consistent lookout than any human operator ever could, whilst also not being susceptible to the mechanical failures as a rotating sensor. Whether a perception system ultimately can be considered to fulfil the expectations of a proper look-out is open to interpretation, as are scenarios that present collision risk.

Sufficient perception coverage for an autonomous inland vessel will be interpreted by this thesis as follows. The first major consideration is the architecture of an inland waterway and the typical navigation paths of vessels along such channels. As discussed during the literature review, inland vessels generally travel along a central path of the channel section in a manner synonymous to travel on road networks. Whilst vessels may travel perpendicularly to this traffic flow, crossover interactions are limited significantly by the channel width and are far less frequently encountered. In addition, the vessels that routinely travel across the channel section tend to be small and manoeuvrable, requiring them to give way to larger, less manoeuvrable vessels travelling down the channel.

As this navigation system is being configured for the Grey Seabax vessel, which can be considered comparable in length-beam ratio to a larger inland vessel such as a freight barge. Focusing perception ahead of the vessel, in its direction of movement where collision risk is higher and where opportunity for avoidance is present can thus be considered a better utilisation of the sensor. The chosen camera configuration can be visualised by the mounted hardware in Fig. 4.2. A total of four Intel Realsense D345i depth devices are used which are each orientated to cover a forward weighted coverage around the vessel. These devices have a depth range of ten metres, which is sufficient for the scale vessel in question, offering a coverage of magnitude over seven times the vessel length. For larger scale applications, device selection or custom stereovision setups would need to consider the specific requirements of the vessel scale, with the imaging sensors and the baseline influencing the range of a stereovision sensor. The baseline refers to the distance between the stereo imaging sensors which influences the disparity region where depth can be interpreted and the imaging sensor itself influences the field of view and data density.

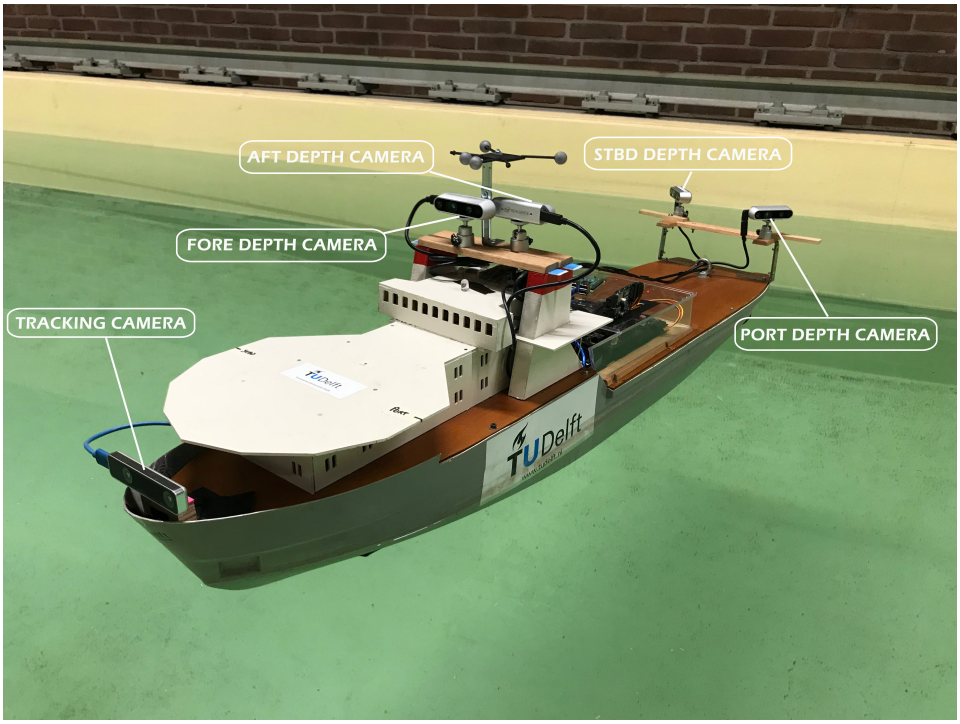
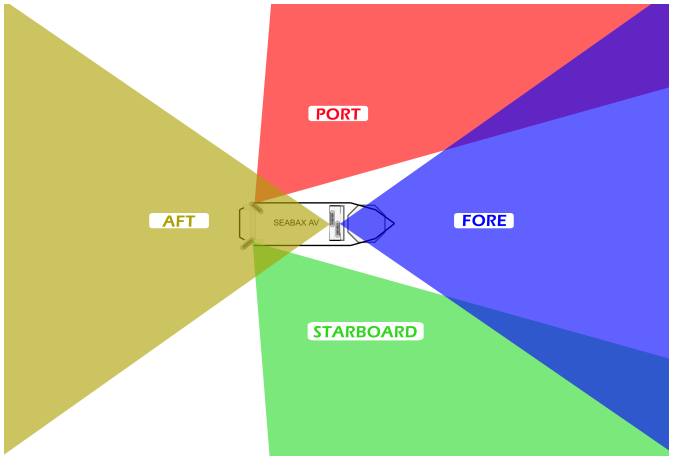


Figure 4.2: Sensor Setup - Grey Seabax Autonomous Test Vessel

The Realsense depth modules utilised onboard the Grey Seabax utilise two infrared sensors and an infrared projector to gather depth data, which offer accuracy even in scenes with low texture. These sensors have a baseline of fifty millimetres and a horizontal field of view covering approximately ninety degrees, equating to a depth map of the same coverage. The RGB sensor within the module offers a reduced horizontal field of view of 69.4° and is not physically aligned with the depth field as can be seen in Fig. 4.3a [46]. Consequently the colour frame and depth frame received from the device must be subjected to alignment during post-processing, resulting in the devices being mounted in such a way to centralise about the colour sensor frame. The resultant coverage around the vessel can be seen in Fig. 4.3b with the horizontal field-of-view for each device being separated by colour and textually labelled in the figure. The exact locations and orientations of cameras on board the vessels will be further disclosed in Section 4.4.1 with a technical drawing.

(a) Intel RealSense Depth Module Configuration [46]



(b) Multi-device Depth Camera Setup

For localisation, a dedicated tracking device is utilised which makes use of two fish eye lenses and a built in inertial measurement unit to conduct stereo Visual- Simultaneous Localisation and Mapping (V-SLAM). Although SLAM algorithms could also be run using data from the perception devices, offloading this task to an external

device drastically reduces computational load on the on-board processor. Furthermore, the dedicated sensor set incorporates a sophisticated list of built in features and considerations which shall be discussed in detail further on in this chapter. The limitations of the technology will also be discussed, which also has some impact upon how the device is mounted to the ship. Intel recommend that the camera should be mounted rigidly, yet should make use of dampened mounts and data cable screws in any case where the device may experience vibrations and/or sudden knocks. In addition, care should be taken to minimise occlusions to the vision sensors field of view as this can prove detrimental to the localisation algorithm and cause significant error.

After testing the performance at various locations, the selected mounting point onboard the Seabax is situated at the bow of the vessel as can be seen in Fig. 4.2 using a 3D printed mount. This mounting location ensures that the vessel itself does not occlude the visual frames and during testing did not appear to be impacted by a large field of view being occupied by the water surface. There was a concern that a lack of features on the surface and/or reflections may have caused total localisation failure, however this was not the case, provided the remaining environment contained enough features. Testing the device when mounted directly to the Seabax also did not demonstrate any of the adverse effects that can be expected from vibrations/knocks, suggesting that the rigid mount is sufficient on board the test vessel without further dampening.

4.2. Perception

When perceiving the environment, a human controller sub-consciously identifies obstacles and approximates their position and state. Artificially replicating this behaviour to a comparable level is extremely complex and arguably an impossible task, yet this remains an open research topic of significant focus within the robotics community. The approach taken in this project to achieve perception will be broken down into three sub-tasks which can be visualised in the procedural system diagram in Fig. 4.4. These tasks of data acquisition, obstacle detection and obstacle analysis form the sub-section structure of this section and cover the approach taken to process depth camera data into an output detailing the instance of obstacles perceived and their respective attributes.

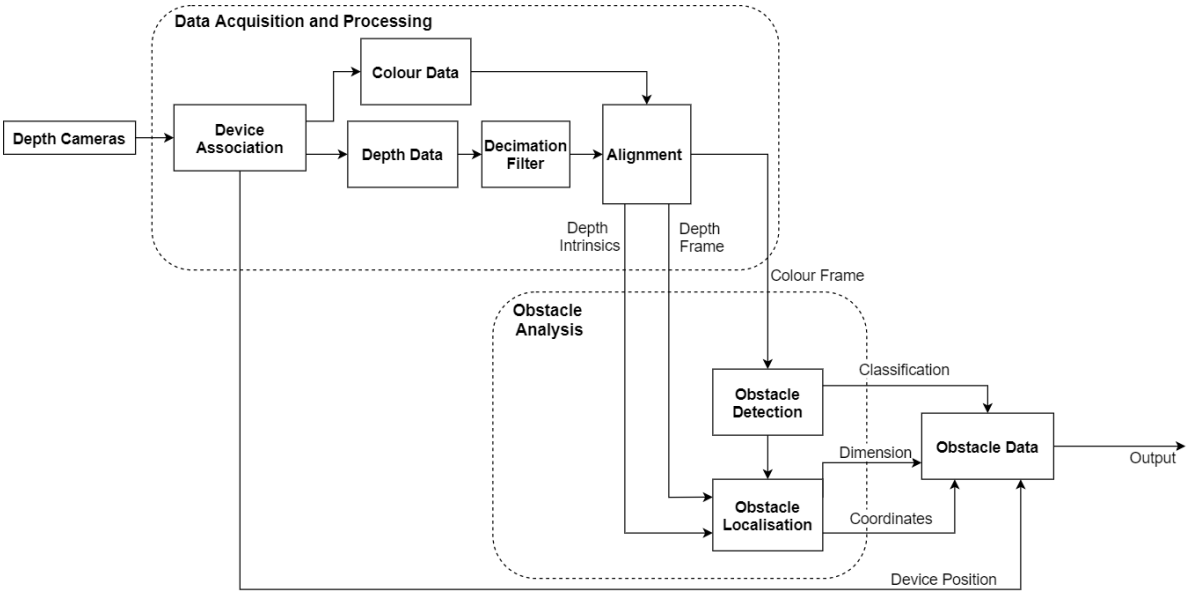


Figure 4.4: Perception sub-task procedure

4.2.1. Data Acquisition and Processing

Data acquisition and processing handles the communication between the perception sensors and the Navigation System and processing the raw data prior to outputting the frame and intrinsic data relevant for ongoing tasks. As this data may wish to be accessed by more nodes than the obstacle analysis task, the sub-task itself runs independently of the other perception tasks. Within the perception ROS package introduced in the previous chapter, this procedure can initialised by running the 'stream_pipeline' executable. The Librealsense Software Development Kit can be accessed via a Python package and includes a number of useful functions for this procedure.

Acquiring data from the perception sensors starts with the configuration of the devices and the relevant data streams. The device serial numbers are primarily associated with their stored location on the vessel so to simplify onward referencing. Two sets of frame data are required from the each device, specifically a colour frame is used for the following obstacle detection task and a depth frame which is required to analyse the actual position of detected obstacles. Fortuitously, these frames can be requested directly from the device as they are processed by the onboard D4 processing chip. When received from the device, the colour and depth frames are not aligned to one another, which is necessary to associate the detected obstacle bounds in the colour frame to the relevant depth data. This forms the first major post-processing task and also proved to have an influence upon the dimensions of the frame data.

Aligning the depth frame with the colour frame can be achieved with a function built into the Librealsense SDK. Albeit a useful tool, this alignment task is however rather computationally heavy meaning that care needs to be taken when employing the function. Whilst Intel has optimised the function performance for use for their own chipsets, the process proved to be significantly slower for the host Jetson chipset for which the perception system operates, even when building the SDK with CUDA capabilities to allow for GPU offloading. That being said, GPU power is in short supply due to the obstacle detection process which runs synchronously with the stream acquisition, not to mention the multi-cam setup leading to overheads being four-fold. The alignment process was further found to be particularly susceptible to any latency within the external data acquisition loop, meaning that streaming other processes synchronously proved sub-optimal. Even by overriding Python's global interpreter lock through multi-processing/threading comes to little avail, most likely due to the SDK reacting unfavourably to the manual assignment of computational power.

There are a few ways in which these issues can be reduced and the performance of the alignment task can be improved. Primarily, it is wise to consider ways to minimise the computational load of this process where possible to ensure that this does not prove to be a bottleneck to the perception program and lead to unsatisfactory latency. The first way that the computational strain of this process can be reduced is through the selection of a lower resolution and frame rate than that recommended by the manufacturer (848x480, 30FPS) [47]. As the application does not require exceptionally high data frequency nor point cloud density due to low sailing speeds and a generalised localisation procedure, the depth frame could be set to a resolution of 424x240 and a frame rate of 15FPS without impacted onward results. Furthermore, as the obstacle detection program resizes any input image to 300x300 pixels, selecting a resolution of 424x240 here also does not lead one to anticipate a reduction in detection performance over higher resolutions. The relative effect of this reduction in resolution and frame rate accounts for a computational strain eight times lower than that of the recommended stream setup.

Furthermore, by applying a sub-sampling post-processing to the depth frame prior to alignment reduces the computational load. Decimating the depth frame through factor of two reduces subsequent computation by a factor of four and has the added benefit of smoothing the depth data, through the removal of dead depth pixels that do not carry a value as they did not reach the confidence metric of the stereo algorithm. Whilst additional processing such as spatial and temporal filters can be implemented to improve point cloud quality, the improvement is unlikely to benefit this application and would come at unnecessary computational cost.

To reduce the impact of the aforementioned loop latency on the alignment task and provide frame data topics accessible by any subscriber in the network, the stream acquisition program is conducted in parallel to the subsequent obstacle detection and analysis processes. In order to create a data pipeline between the programs being executed in parallel, the ROS communications protocol is utilised using a node to publish data with two topics existing per enabled device. One for the colour frame and the second for the (aligned) depth frame. The depth intrinsics of each device is also required for ongoing processes, however this does not change during streaming and so is saved to a file when the devices are first initiated and can be accessed by any program on the Jeston. The intrinsics contain data on the depth calibration and format of each of the depth device frames and will be accessed by the obstacle analysis program. The ROS node initiated by this program consequently publishes eight topics to the network, with each publisher being generated under the name of frame type and the location of the device for which they are associated as can be seen in Table. 4.1.

As the Jetson does not function as the ROS master, the bandwidth capacity must be sufficient to handle the streaming of all four devices over the ROS network. Each device streams colour (8-bit) and depth (16-bit) frame data over the network at 15 frames per second and with a resolution of 424x240. Summing the raw data throughput for all outgoing streams results in 36.6 Mb/s, which even when allowing contingency for encoding overheads can easily be handled by a WiFi router. Encoding is managed using the CV bridge ROS package which allows image data to be published over the ROS network and decoded at the other end by the subscriber.

Device Position	ROS Topic Name	
	Colour (uint8)	Depth (uint16)
Fore	/ForeColour	/ForeDepth
Aft	/AftColour	/AftDepth
Port	/PortColour	/PortDepth
Starboard	/StbdColour	/StbdDepth

Table 4.1: ROS Topics published by Pipeline Node

4.2.2. Obstacle Analysis

With sensor data being processed into a suitable form, perception can direct its focus analysing the environmental state for the instance of obstacles and determine their position. Figure 4.4 provides an insight to the obstacle analysis sub-system where the tasks of obstacle detection and localisation are divided. Obstacle detection is conducted using the colour frames acquired from each of the devices, with the classification of the obstacle being supplied directly as an output of the analysis sub-system and the obstacle bounds being provided as an input to the obstacle localisation task. Each obstacle bound defines a rectangular area within the frame where an obstacle has been detected and due to frame alignment, this Region-of-Interest (ROI) also corresponds to the same frame area within the depth frame. Using the device intrinsics to assess the ROI within the depth frame allows for the obstacle location to be determined, along with a horizontal obstacle dimension. The selected procedure for obstacle detection and localisation is discussed in specifics hereon.

4.2.2.1. Obstacle Detection

Obstacle detection is a critical primary task for any perception system. The most popular technique for conducting object detection for autonomous vehicles sees the use of clustering techniques applied to point cloud data. However the emergence of Artificial Intelligence based object detection on colour images has grown in popularity of late, with both 2D and 3D object detection tools being developed using convolutional neural networks. One major benefit of the artificial intelligence approach is the ability to classify objects beyond arbitrary detection. Knowing the type of obstacle that has been detected allows a guidance system to make better informed decisions on collision avoidance.

The Jetson TX2 Development Kit has a small form factor making it ideal for use as an onboard processing unit. Despite its small footprint, the board exhibits outstanding performance when running inference on a neural network model due to its on board graphical processor being especially well suited to Artificial intelligence tasks. To fully utilise the potential processing power of the TX2, it is beneficial to convert trained neural network models into TensorRT engines. TensorRT is a software development kit provided by NVIDIA for use with their hardware to allow developers to fully optimise their AI applications.

Format	Data Type	Speed (ms)	mAP@IoU=0.5:0.95
Tensorflow	FP32	125	24.6
TFLite (Quantised)	INT8	143	19.0
TensorRT Engine	FP16	25	24.6

Table 4.2: Comparison of Mobilenet SSD v2 Model Performance using Speed and Mean Average Precision (mAP @IoU=0.5:0.95) as Key Performance Indicators

When assessing the performance of TensorRT, it was compared with two other frameworks, Tensorflow and TFLite. TFLite had been implemented in a previous RAS project where the Jetson was not available [48]. When running inference on a CPU bound PC, utilising the TFLite framework with an 8-bit quantised model provided optimality regarding inference speeds, albeit at the cost of accuracy. As the Jetson is optimised for GPU processing, there is little advantage in running with the integer TFLite framework as can be observed in Table 4.2, where both speed and accuracy are proved sub-optimal compared to the standard Tensorflow framework. Both speed and accuracy of inference can be improved by switching to a floating point based model. To quantify this performance upgrade, TFLite inference on the Jetson yielded speeds of approximately 7 frames per second and when running inference of the same MobileNet SSD v2 COCO model in the through a TensorRT engine, the frame rate jumped to approximately 40 frames per second. Furthermore when evaluating the mean average precision (mAP) achievable using the val2017 dataset, the TensorRT engine provided the equivalent precision to the baseline Tensorflow model, despite its reduction from a 32 bit to a 16 bit model.

Using a TensorRT engine for running inference consequently offers the best option when considering both speed and accuracy. Even when considering the multi-device perception configuration, as inference only takes 25ms per frame (40Hz), running detection on all four cameras streams could theoretically allow for near 10Hz per camera frame. A COCO model, whilst useful for testing and development cannot be used to detect the vessels in the simulated test environment, meaning a new model needed to be trained for the purpose.

In a fully-fledged application, the neural network model would need to recognise all obstacles that may be pose a collision risk in the inland waterway environment. At this stage of the development however, the model will only be trained to recognise one object, specifically the model tugboat that will be used in the experimental collision avoidance testing. The training of such a Convolutional Neural Network model is covered in the paper preceding this work [48] which details the steps required to obtain a standard Tensorflow model.

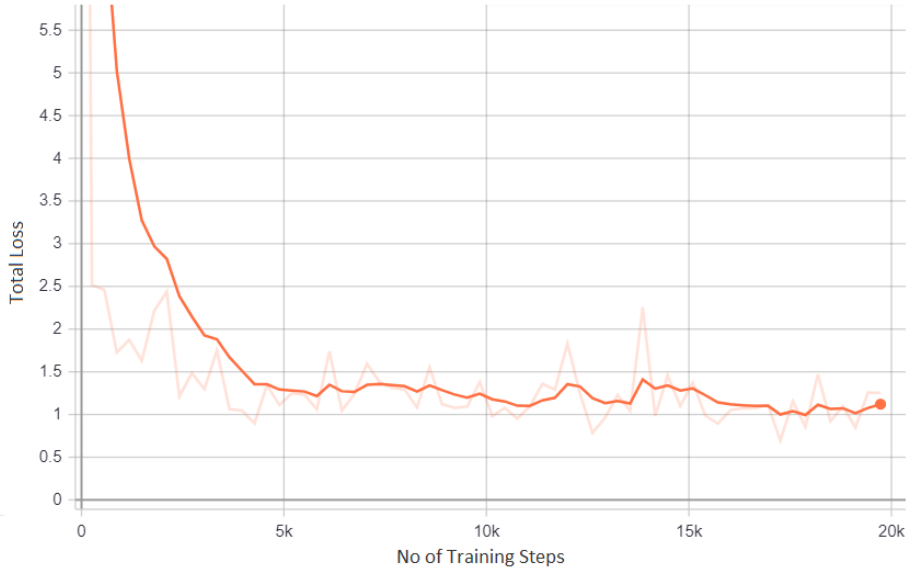
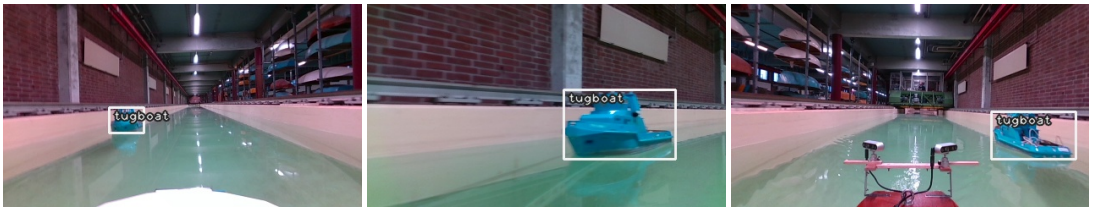


Figure 4.5: Average Total Loss relative to Training Iterations

The MobileNet SSD v2 model was trained using approximately 1200 annotated images of the model tugboat within the testing tank environment. The training results for this model are displayed in Fig. 4.5, where the graph shows a steady reduction of total loss until levelling off at an average loss of 1.2 after approximately twenty thousand training iterations. This performance metric effectively quantifies the discrepancy between the model result and the actual result, the aim is consequently to minimise this discrepancy until convergence occurs. With a trained model, the standard Tensorflow format must then be converted to a Universal Framework Format (UFF) and subsequently parsed to build an optimised TensorRT engine.

Detection results can be monitored whilst the task is running, by observing the graphical interface for each camera feed. Figure 5.3 shows example outputs for multiple camera positions demonstrating the detection ability from all views surrounding the vessel. As can be seen, each detection is highlighted by the annotation of the object bounds and classification. This user output provides not only an excellent tool for monitoring detection but also for development and evaluation of the neural network performance.



(a) Fore colour frame (b) Side colour frame (c) Aft colour frame

Figure 4.6: Visualised detection output at various device positions.

4.2.2.2. Obstacle Localisation

The obstacles detected in the colour frame must subsequently be analysed to determine their actual position in the 3D space. In maritime surface applications, the vertical positioning of an obstacle is of little importance as the vessel cannot move in this direction. Horizontal and depth positioning is however of critical importance as these are within the degrees of freedom of the vessel which it must control. During this stage of obstacle analysis, only the position of the obstacle relative to the respective device is calculated. The positioning of obstacles will later be translated into one homogeneous reference frame during the mapping stage as detailed in 4.4.

Prior to analysing the obstacle position, it is first checked whether any occlusions exist in the bounds of the detected obstacles as this can significantly influence the depth value that is obtained within the individual obstacle bounds. As detection is conducted on a 2D representation of the environment, multiple obstacles may be detected in the same pixel range leading to all obstacles being associated with the depth data of the nearest object. To reduce the likelihood of this error occurring, all the detected objects in each frame are checked for any overlap in their bounding boxes using the procedure detailed in Algorithm **??**. In the case where an occlusion is detected, instead of using the bounding box of the affected obstacles for extracting depth data, a new region of interest (ROI) is defined. This new ROI retains the same centroid as the original bounding box, however reduces its size to one third of the original. By limiting depth data extraction to this innermost region helps remove errors caused by partial occlusions. The dimensions of these bounds were determined through experimental testing of the object detection model which showed that bounding boxes may overlap, however once an object was occluded by more than a third of its area, it could no longer detected by the model.

The distance to the obstacle in terms of depth from the camera reference plane is determined by assessing all the depth points within the obstacle bounds and selecting the lowest value. The reason for selecting the lowest value is two-fold, primarily as a safety concern and also regarding data consistency. The safest assumption is that the closest part of the obstacle is the most likely point of collision and this value also provides the most consistent depth result between frames. If instead of the lowest depth value, the mean average of all depth points within the region are taken, there exists the risk of causing an unexpected collision due to underestimating the obstacles position. The mean depth result also proves to be more volatile over time as the mean average fluctuates far more than the lowest value when obstacle orientation changes. This selected depth point shall be considered as the Closest Obstacle Point (COP).

Immediately prior to finding the lowest depth value within the ROI, misleading data must first be purged from the array. When the RealSense device runs the stereo algorithm, any pixel which could not be assigned a depth value due to confidence metrics not being met, is assigned the value of zero to indicate this error to onward processes. Therefore, all the zero values must be removed from the depth array prior to selecting the lowest value, otherwise any ROI where a zeroed pixel resides will result in the obstacle being associated an invalid depth value. Furthermore, the field of view of the fore and aft cameras are actually occluded slightly by the vessel itself. This could also lead to misleading depth values and to avoid this, the depth arrays of the fore and aft cameras are further filtered to remove any depth values that lie within the vessel bounds. For the aft device, this filter is set to a depth of one metre and for the fore device, 0.4m.

Whilst there exists the possibility to create a detailed profile of an obstacle with the point-cloud data, it is both computationally expensive and arguably unnecessary for typical collision avoidance scenarios. It may be argued that a detailed point-cloud could assist in the estimation of vessel heading, however such a task would be highly complex, computationally expensive and have a large margin for error. As is further detailed in the following Chapter, the heading of a contact vessel will not be used in the collision avoidance approach, rather the vessel's track shall be used to record and predict motion. The track of the vessel will be interpreted later during the mapping process, see section 4.4.

With a depth value obtained, the lateral location of the obstacle and its bounded width can also be found. For simplicity, lateral reference point for the vessel will be taken as the centroid of the obstacle bounding box. The depth intrinsics for each device must at this stage be recalled so to project the pixel values into 3D coordinate points. To convert to the lens coordinate frame, the principal point is subtracted and the sum is divided by the focal length, following which the impact of specific lens distortion is accounted for. Fortuitously, this projection can be achieved using a built in Librealsense function, which requires only the depth intrinsics, pixel of interest and the depth value to be supplied. For each obstacle two points are projected to 3D coordinates using the obtained depth value and the pixel values at of the horizontal extremities of the bounding box. By extracting these two points, the width of the obstacle can subsequently be estimated and the lateral position of the obstacle centroid easily derived. As the obstacle orientation is unknown, the width of the bounding box provides only an

arbitrary obstacle dimension.

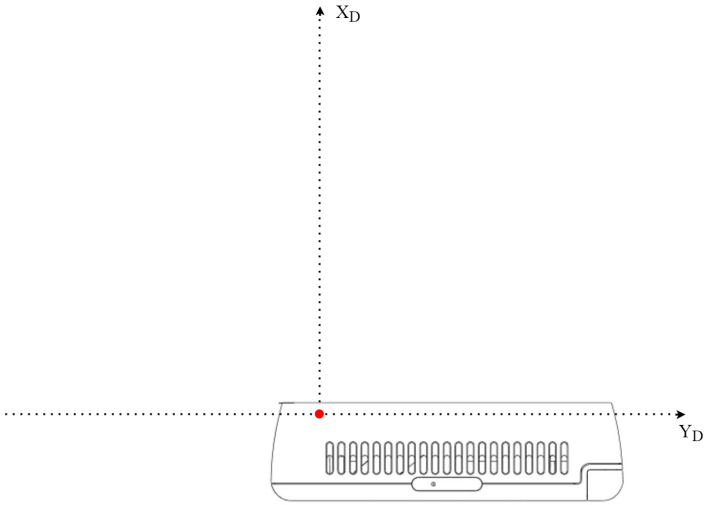


Figure 4.7: Definition of the Device Coordinate System - *Diagram displays a plan view perspective of the D435i Depth Camera with the origin of the coordinate system incident to the RGB sensor.*

Obstacle position is then defined defined within a device coordinate system (X_D, Y_D) as shown in Fig. 4.7, where the depth value corresponds to the X_D axis and the lateral position of the obstacle is to the Y_D axis. The origin of the depth is at the centre the left depth imaging sensor and the origin of the lateral distance is centred about the horizontal mid-plane of the colour frame (aligned depth frame). Whilst depth values can only be positive, points to the left of the mid-plane are assigned a negative coordinate value and points to the right, a positive value.

4.2.3. Perception Output

The output from the perception package is published over the ROS network as a string message at a frequency of 5Hz. The contents of the message contains all the relevant information for the ongoing processes and are configured in such a manner to allow the subscriber to simply convert the message into a Python dictionary. Firstly an obstacle count is provided which states how many obstacles have been detected by the perception program across all the devices. Following this, each of the obstacles and their corresponding attributes are listed, with each obstacle being an item within the list. The attributes that are published for each object are as follows:

- Device position (i.e. 'Fore'/'Aft'/'Port'/'Starboard')
- Obstacle classification (e.g. 'Tugboat')
- Obstacle coordinates ((X_D, Y_D))
- Obstacle dimension

4.3. Localisation

Localisation concerns the determination of the Autonomous Vessels current position and orientation in the configuration space. Typically a global positioning system (GPS) provides the primary data for localisation, however as the experimental testing will take place indoors, an alternative solution is required to ensure sufficient readings. Fortunately, stereovision also offers a solution to localisation through the implementation of V-SLAM algorithms. As such as well as reviewing the potential of stereovision as a perception sensor, this paper will also review the viability of using stereovision sensors to gather localisation data. Localisation accuracy is of critical importance to the performance of an autonomous vessel and this can be best achieved by using multiple sensors for acquisition and applying data fusion. By assessing the performance of stereovision for the specific use case onboard an inland autonomous vessel, its potential for future application on larger scale autonomous vessels can be understood. The accuracy level of stereovision could complement GPS data nicely, with GPS alone being limited in accuracy and impacted by occlusions in its direct view of the sky. The restricted channel widths on inland waterway heightens the requirement for accurate localisation and occlusions to the sky such as bridges could further reduce GPS accuracy.

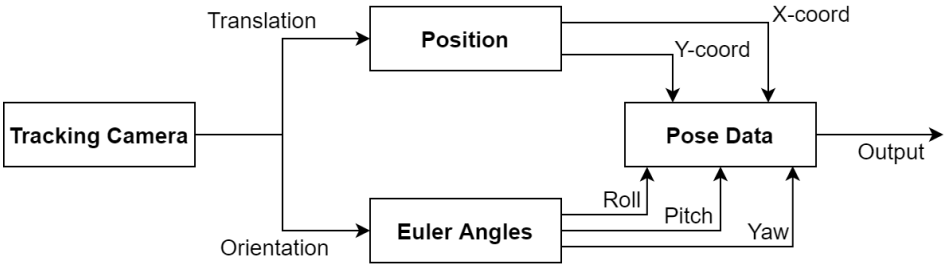


Figure 4.8: Localisation Procedure

Figure 4.8 provides an overview of the localisation sub-system from the sensor input through to the sub-system's pose output. As the device executes the localisation algorithm on its built-in processor, the data acquired directly from the device comes in the pose format of translation and orientation. The following sub-section shall detail the background concerning the onboard algorithm, its features and restrictions. The sub-section thereafter will cover the translation of the raw pose data from the device into the desired configuration. The reference frame of the device is not aligned with that of the vessel, meaning that the position needs to be derived by axes transformation and the orientation must be calculated from quaternions into Euler angle form.

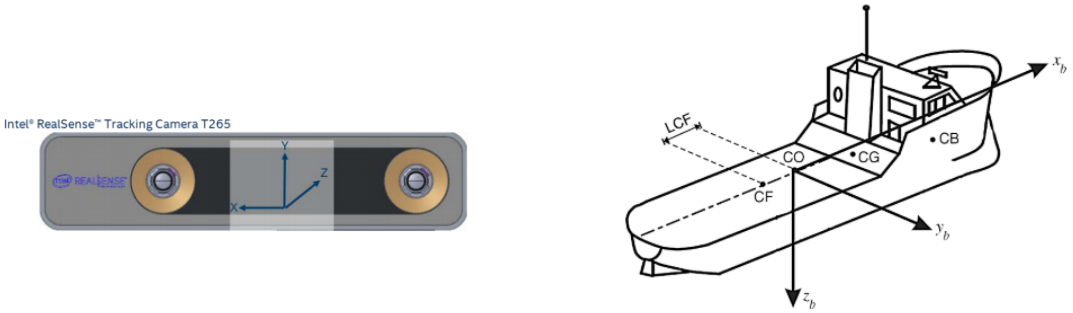
4.3.1. Background

Although data from the depth cameras could provide data input to a point cloud based V-SLAM algorithm, the device sensors are configured for optimal depth accuracy not to provide optimal data for localisation. The T265 tracking camera however, is optimised for conducting localisation using a Visual-Intertial-Odometry (VIO) technique. The principles behind VIO are akin to that of human localisation, where key features in a scene are recognised and tracked over time, along with inertial movements. The T265 uses computer vision algorithms to recognise features captured by two fish-eye lenses which can be compared to the visual data obtained by human eyes and an Inertial Measurement Unit to record movement and rotation, the same task managed in the inner ear of a human.

The use of two imaging sensors (stereo) not only increases the amount of features that can be recognised, but further provide a calibrated method for scaling the environment into real world coordinates. Whilst modern imaging sensors can operate at high frame rates, the algorithms that are used detect features in each frame require high computational power to keep up with such frequencies. IMU data on the other hand can be supplied at high data frequencies without requiring laborious post-processing. This high frequency characteristic makes IMUs extremely good at tracking movements over short time periods, to the degree of a fraction of a second. The data from the imaging sensors and IMU subsequently complement one another well, allowing for low latency tracking to be achieved without huge computational expense. The device in fact executes the VIO processing using its onboard chip, requiring only 1.5W via a USB data cable, entirely offloading the burden from the host board [49].

The device also minimises the impact of sensor drift error by providing re-localisation contingency to centimetre accuracy. In the case where the device momentarily loses its bearings, the device reviews its memory of the environment to best relocate itself based upon known features within the map. Whilst the potential benefits of stereovision based localisation have been made apparent, a few limitations of the technology are worth noting.

Operating in extreme light conditions, whether that be low or high can have an impact on the quality of localisation. Too many moving objects in the environment, or an environment with two few landmark features can lead to significant localisation errors as there are no clear stationary features to track between frames. The imaging sensors are further limited in the same way as humans, where they can neither see in the dark nor when facing a direct light source. Reflections of light can also cause issues, which may be a particular issue in the maritime application with the still water surface potentially causing phantom features to be recognised. Whilst the mounting solution used for this project does lead to a discouraged occlusion in camera field of view, the surface of the water does not provide any useful features for tracking and avoiding a direct view of the water from an acute angle, may reduce the likelihood of reflection errors. Extra attention should be paid during experimental testing to these areas of potential error occurrence.



(a) Tracking Device Coordinate System [50]

(b) Body-fixed Coordinate System [51]

Figure 4.9: Localisation Coordinate Systems

4.3.2. Translation

Acquisition of the localisation data from the device is again achieved using the Librealsense SDK through the Python wrapper. The Python executable responsible for the acquisition and manipulation of localisation data can be found within the 'ras_localisation' ROS package as seen in Fig. 3.3. The coordinate system of the tracking camera itself is defined in Figure 4.9a with the centre of tracking equidistant from the two imaging sensors. This coordinate system differs from the orientation of the body-fixed coordinate system and the North East Down (NED) reference frame and so must be translated for ongoing processes. The raw tracking data is translated to the body-fixed frame of the vessel itself, see Fig. 4.9b, meaning the X and Y coordinate (x_b, y_b) that is finally published by the ROS package refers to the negated z and x values of the tracking device coordinates by applying the transformations defined in table 4.3.

NED	Body-fixed	Tracking Device
North	x_b	-z
East	y_b	x
Down	z_b	-y

Table 4.3: Coordinate System References

The Euler angles roll, pitch and yaw have to be calculated from the raw pose data which comes in the form of a quaternion. The quaternion format offers a far less intuitive representation than Euler angles, however are advantageous as they are unrestricted by the gimbal lock which prevents measurements when pitch reaches ±90°. In nautical applications, a vessel should never reach such extreme pitch rotations and so conversion to the Euler angles can be conducted without further consideration to the impact of gimbal locking.

A quaternion is a vector comprised of four elements, one real and three imaginary, which can be used to describe the orientation of a body in a three dimensional environment. The quaternion data from the tracking device is delivered in the order x, y, z, w. Where w is the real part and x, y, z represent the imaginary vector parts. Aligning the unit quaternions to the body frame can be achieved by again implementing the relations seen in Table 4.3. As the conversion to Euler angles will follow the procedure of Fossen et al. [51], the unit quaternions will be further converted to the relevant notation seen in Eq. 4.1.

$$q = \begin{bmatrix} w & -z & x & -y \end{bmatrix}^T = \begin{bmatrix} \eta & \epsilon_1 & \epsilon_2 & \epsilon_3 \end{bmatrix}^T \tag{4.1}$$

The coordinate transformation matrix for the unit quaternions can be defined by Eq. 4.2 which enables conversion to the Euler angles of roll, pitch and yaw using Equations 4.3, 4.4 & 4.5 respectively.

$$R_b^n(q) = \begin{bmatrix} 1-2(\epsilon_2^2+\epsilon_3^2) & 2(\epsilon_1\epsilon_2-\epsilon_3\eta) & 2(\epsilon_1\epsilon_3+\epsilon_2\eta) \\ 2(\epsilon_1\epsilon_2+\epsilon_3\eta) & 1-2(\epsilon_1^2+\epsilon_3^2) & 2(\epsilon_2\epsilon_3-\epsilon_1\eta) \\ 2(\epsilon_1\epsilon_3-\epsilon_2\eta & 2(\epsilon_2\epsilon_3+\epsilon_1\eta) & 1-2(\epsilon_1^2+\epsilon_2^2) \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \tag{4.2}$$

$$\phi = atan2(R_{32}, R_{33}) \tag{4.3}$$

$$\theta = -asin(R_{31}): \quad \theta \neq \pm 90^\circ \tag{4.4}$$

$$\psi = atan2(R_{21}, R_{11}) \tag{4.5}$$

4.3.3. Localisation Output

Once the relevant information has been obtained, it is published over the ROS network as a string message, ready for ongoing processes to subscribe to the stream and again can be easily converted into a Python dictionary. The following data is obtained and published: X coordinate, Y coordinate, Roll, Pitch and Yaw. Whilst ongoing processes only require position and yaw directly as this informs as to the current coordinate of the Autonomous Vessel and of the current heading. The other data is published due to its potential use in future projects. The yaw value is also published in its own topic as it will be subscribed to directly by the control system to provide an input to its heading error strategy.

4.4. Mapping

Mapping represents the final sub-task in the Navigation system and is responsible for the post-processing of the results from the perception and localisation sub-tasks. This processing concerns two main procedures, firstly the translation of perception results to a single homogeneous reference frame, and secondly the fusion of localisation and perception data to map the position of the AV and obstacles in the global coordinate system. Subsection 4.4.1 details how the obstacle positions attained by the individual perception sensors are translated to the coordinate system of the Autonomous Vessel. Subsection 4.15 details the generation of a global map, where both the AV and the obstacles are plotted in the global reference frame.

4.4.1. Autonomous Vessel Domain

The initial stage of mapping handles the translation of the obstacle positions from the individual device coordinate system into the coordinate system relative to a consolidated AV domain. The device coordinate system is defined as having two axis relative to the camera, where X_D is aligned with the depth value of the camera and Y_D refers to the position in the horizontal frame. The origin of the AV domain, is located at the geometric centre of the vessel as indicated by point CO in Fig.4.9b and reiterated in Fig. 4.10, with the axes orientation of x_b and y_b also matching that of the 2D AV coordinate system, X_{AV_D} and Y_{AV_D}. When mapping obstacles to the AV domain, each instance will be assigned with a 2D coordinate relative to this origin and reference axes.

As each device is positioned at a different location on the vessel, with a different orientation, there exist varied procedures for generating the AV domain coordinates. Fig. 4.10 details the locations and orientations of each depth device with the relevant dimensions needed for the translation to the AV domain. As can be observed in

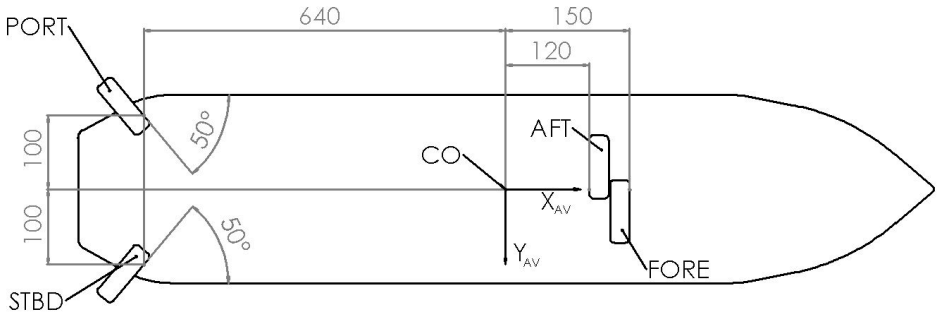


Figure 4.10: Multi-device Configuration for Perception. *All linear dimensions in millimetres.*

the diagram, translation of the fore and aft, is fairly straight forward as the orientation and x-axis origin is already aligned with the AV coordinate system only the y-axis displacement must be displaced.

The port and starboard cameras on the other hand are offset from the origin in both the axes and are further orientated at $\gamma = \pm 50°$ displacement from the AV coordinate system. Therefore, prior to accounting for the displacements in the axes, the obstacle coordinates must be translated by rotating the device axes to reach a parallel relation with the AV coordinate axes. To translate the device coordinates (X_D, Y_D) to be parallel with the AV domain axes(X'_D, Y'_D), the axes rotation defined in 4.6 is performed. After the axes have been aligned, the axes displacements from Fig. 4.10 can be applied as detailed in Algorithm 1 to define the obstacle's coordinates in the AV domain.

$$\begin{pmatrix} X'_D \\ Y'_D \end{pmatrix} = \begin{pmatrix} cos\gamma & sin\gamma \\ -sin\gamma & cos\gamma \end{pmatrix} \begin{pmatrix} X_D \\ Y_D \end{pmatrix} \tag{4.6}$$

Algorithm 1: AV Domain Displacements

Result: Obstacle Coordinates in the AV Domain
if *Device is 'Fore'* **then**
  $X_{AV_D} = X_D + 0.15$
  $Y_{AV_D} = Y_D$
else if *Device is 'Aft'* **then**
  $X_{AV_D} = 0.12 - X_D$
  $Y_{AV_D} = -Y_D$
else
  if *Device is 'Port'* **then**
    $X_{AV_D} = X'_D - 0.64$
    $Y_{AV_D} = Y'_D - 0.1$
  else
    $X_{AV_D} = X'_D - 0.64$
    $Y_{AV_D} = Y'_D + 0.1$
  end
end

With the obstacle coordinates defined in the AV domain, the bearing (β) of the obstacle relative to the AV can be derived. To do this, the angle between the origin x-axis of the AV domain and each obstacle needs to be found. Where the origin coordinates are (X_{CO}, Y_{CO}) and the range is r, the obstacle coordinates are (X_{AV_D}, Y_{AV_D}) can be defined as shown in Eq. 4.7 and the raw bearing angle $\hat{\beta}$ in Eq. 4.8.

$$(X_{AV_D}, Y_{AV_D}) = (X_{CO} + r \cdot \hat{\beta}, Y_{CO} + r \cdot sin\hat{\beta}) \tag{4.7}$$

$$\hat{\beta} = atan2(Y_{AV_D} - Y_{CO}, X_{AV_D} - X_{CO}) \tag{4.8}$$

As the origin coordinates are (0, 0), Eq. 4.8 can be simplified to Eq.4.9 and the actual bearing β can be defined by the cases in Eq. 4.10.

$$\hat{\beta} = atan2(Y_{AV_D}, X_{AV_D}) \cdot \frac{180}{\pi} \tag{4.9}$$

$$\beta = \begin{cases} \hat{\beta}, & \hat{\beta} \geq 0° \\ 360 + \hat{\beta}, & \hat{\beta} < 0° \end{cases} \tag{4.10}$$

Referring once more to Fig. 4.3b, it can be observed that an overlap in field-of-view that exists between the fore and side devices and this region can lead to the perception sub-system essentially seeing double when the same obstacle is recognised by multiple cameras. It is important that these replicated obstacles are removed so to ensure that ongoing processes do not consider these obstacles to be independent. This is achieved following the procedure in Algorithm 2, whereby obstacles who share classification and almost identical coordinates are removed from the obstacles list.

Algorithm 2: Obstacle Replication Removal

Result: Replication Removal
for *i Fore_obstacles* **do**
  for *j in Port_obstacles and Stbd_obstacles* **do**
    Vicinity = $|coords[i] - coords[j]|$
    if *Vicinity* $\leq$ *0.5m and classification[i]==classification[j]* **then**
      Remove(j)
  end
end

With the obstacles having been coalesced into a singular coordinate frame and any replications eradicated, they can each be associated with an identification number for tracking between iterations. As obstacle detection is run afresh each frame, the obstacle log generated with each iteration is arbitrary and bears no direct association to the previous set of detections. Tracking is instead achieved using a Nearest Neighbour technique whereby the coordinates of each obstacle detected in the current iteration is compared with all obstacle positions in the previous iteration and associated by minimising Euclidean distance. This procedure allows for the assignment of an obstacle identification number (ID).

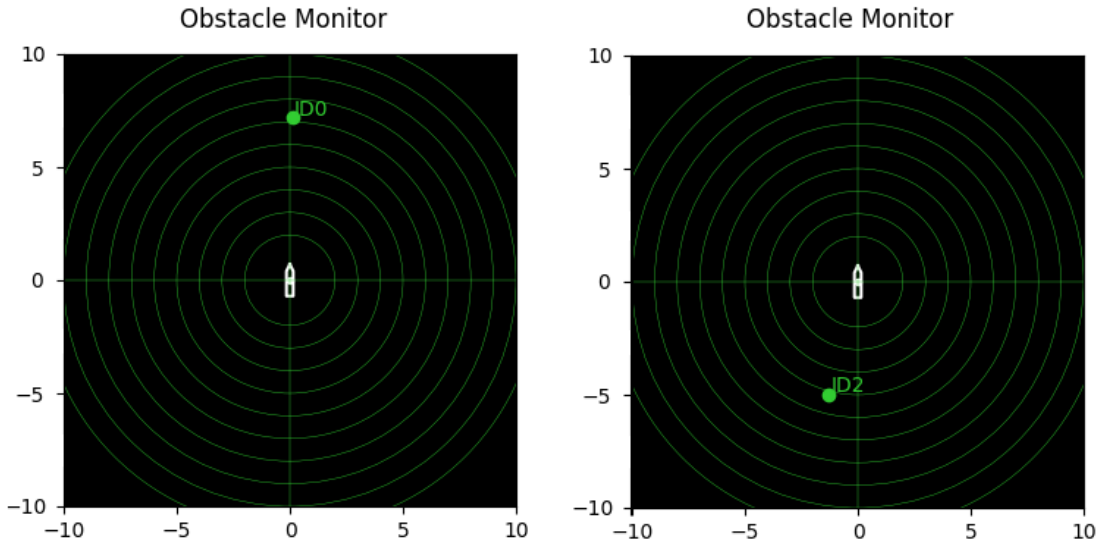


Figure 4.11: Pseudo-radar Obstacle Monitor

To enable user-interaction with the navigation system, a visual representation of the obstacles mapped within the AV domain is provided. In line with the maritime application, this output is delivered in the form of a pseudo-radar plot as can be seen in Fig. 4.11. The origin of the plot corresponds with the origin of the AV domain and is illustrated in the plot by the white dot, with the outline of the vessel plotted around this point for reference. An obstacle that has been detected, localised and mapped is plotted as green dot along with an annotation of its ID.

4.4.2. Global Coordinate System

With localisation data on the Autonomous vessel itself and the surrounding obstacles in the AV domain, global mapping can be initiated to provide the Guidance system with a canvas to track movements and generate trajectories. The global coordinate system needs to be presented with a base map relevant to the area of navigation. For example, in the case of this project, a base map of the tank that shall be used for testing is supplied. As this base map is simply a rectangular section of the testing tank without the presence of static waterway obstacles, it can be specified simply by 4 sets of two dimensional coordinates. This map can be visualised in Figure 4.12 below, along with the defined coordinate system and origin. Outside of this specific application in a testing tank, the base map could be generated using GPS coordinates defining the extremities of the waterways channel.

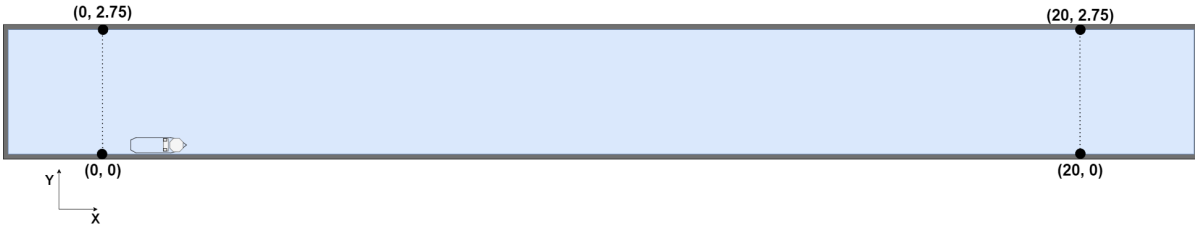


Figure 4.12: Visual representation of the testing tank base map

When first initialising the navigation system, the starting position (X_{av_0}, Y_{av_0}) of the vessel within the base map must be stated. The localisation system is only capable of tracking its movement over time, it does not have the capacity to independently localise its position in the base map. The default starting coordinates for operations in the test tank are (1, 0.15), which describes the location where the vessel is launched for testing. In the case where the start location is different, this must be manually altered prior to initialisation. With this start value, the position coordinates streamed by the localisation system (x_b, y_b) can simply be summed with the starting coordinates to provide the current vessel location in global coordinates (X_{av}, Y_{av}) as per Eq. 4.11. By monitoring this location over time, the state can be quantified by the registered change in position from the previous time step $(\dot{X}_{av}, \dot{Y}_{av})$. A time step of one second is used allowing for the AV speed to be defined by Eq. 4.12 in meters per second.

$$\begin{pmatrix} X_{av} \\ Y_{av} \end{pmatrix} = \begin{pmatrix} X_{av_0} \\ Y_{av_0} \end{pmatrix} + \begin{pmatrix} x_b \\ y_b \end{pmatrix} \tag{4.11}$$

$$u_{av} = \sqrt{\dot{X}_{av}^2 + \dot{Y}_{av}^2} \tag{4.12}$$

As well as a start position, the starting orientation must also be supplied in the form of a initial heading value (φ_0). As per nautical standards, the base map will rely on a magnetic compass to describe vessel heading. For the test tank base map, φ_0 is taken as 250°, which is aligned with the X-axis of the base map coordinates, meaning that when initiated the vessel should be positioned facing directly down the length of the tank. The relative heading value can subsequently be calculated through summing the initial offset with the yaw value (ψ) as defined in Eq. 4.13 and then applying the cases in Eq. 4.14.

$$\hat{\varphi} = \varphi_0 + \frac{180\psi}{\pi} \tag{4.13}$$

$$\varphi = \begin{cases} \hat{\varphi}\,, & 0 \le \hat{\varphi} \le 360^\circ \\ 360 + \hat{\varphi}\,, & \hat{\varphi} < 0^\circ \\ \hat{\varphi} - 360\,, & \hat{\varphi} > 360^\circ \end{cases} \tag{4.14}$$

Defining the global coordinates of each obstacle (X_{cv}, Y_{cv}) is accomplished through the translation of obstacle coordinates in the AV domain using the current AV position and orientation as detailed in Eq. 4.15. Using the obstacle IDs from the previous section, obstacles can be tracked over time, the state of an obstacle can be quantified by the difference in obstacle position at the previous time step at the current iteration position $(\dot{X}_{cv}, \dot{Y}_{cv})$. This time step is again taken to be one second.

$$\begin{pmatrix} X_{cv} \\ Y_{cv} \end{pmatrix} = \begin{pmatrix} X_{av} \\ Y_{av} \end{pmatrix} + \begin{pmatrix} cos\psi & sin\psi \\ -sin\psi & cos\psi \end{pmatrix} \begin{pmatrix} X_{AV_D} \\ Y_{AV_D} \end{pmatrix} \tag{4.15}$$

The velocity vector generated between the obstacle position at the previous time step and the current position can be referred to as the obstacle's relative track. The track vector can be described by its magnitude and direction. The direction defined in Eq. 4.18 being referred to as the Track Angle (τ) and the magnitude describing the vessel speed (Eq. 4.16). To assist in collision avoidance procedure it is also useful to know the angle at which a dynamic obstacle is moving relative to the AV itself. This can simply be determined by finding the difference between the heading of the AV and the track angle of the obstacle. This directional measure of obstacle dynamics shall be referred to as the approach angle (α) and is defined in Eq. 4.20.

$$u_{cv} = \sqrt{\dot{X}_{cv}^2 + \dot{Y}_{cv}^2} \tag{4.16}$$

$$\hat{\tau} = (atan2(\dot{Y}_{cv}, \dot{X}_{cv}) + \varphi_0) \cdot \frac{180}{\pi} \tag{4.17}$$

$$\tau = \begin{cases} \hat{\tau}\,, & 0 \le \hat{\tau} \le 360^\circ \\ 360 + \hat{\tau}\,, & \hat{\tau} < 0^\circ \\ \hat{\tau} - 360, & \hat{\tau} > 360^\circ \end{cases} \tag{4.18}$$

$$\hat{\alpha} = \tau - \varphi \tag{4.19}$$

$$\alpha = \begin{cases} \hat{\alpha}, & \hat{\alpha} \ge 0 \\ \hat{\alpha} + 360, & \hat{\alpha} < 0 \end{cases} \tag{4.20}$$

Of course, these state attributes cannot be generated until the vessel has been tracked over multiple time steps and in this momentary period it is set to a null value to avoid feeding misleading values. The average speed and angle measurements are then taken over the past three seconds so to provide a smoother forward data feed.

4.5. Navigation Output

The output of the mapping process is a refined set of attributes relating to the AV and the obstacles which will form the main output of the entire Navigation System. These attributes and some other will go on to be used by the guidance system during its collision avoidance procedure in the next chapter. To simplify the guidance subscription process, obstacles and all their useful attributes shall be published in one message and the AV attributes shall be published in another. Both of these publishers are set to run at the same frequency and triggered at the same time so to bi-pass the requisite of time-stamping data. The attainable frequency was found to vary somewhat, however could always achieve 5Hz leading to this rate being fixed to ensure consistency in publication. The exact attributes published over the ROS network as the final output of the navigation system are detailed below in table 4.4. The AV position will also be subscribed to by the high level controller of the Seabax control system as an input to the reference heading calculation, which shall be discussed further in Chapter 6. Consequently an additional topic is published, dedicated to providing the current position in array format to the control system.

Attribute	**AV Data**	**CV Data**	**Notes**
Global Coordinates	(X_{av}, Y_{av})	(X_{cv}, Y_{cv})	Eq.4.11, Eq.4.15
AV Domain Coordinates	-	(X_{AV_D}, Y_{AV_D})	Algorithm.1
Speed	u_{av}	u_{cv}	Eq.4.12, Eq. 4.16
Heading	φ	-	Eq.4.14
Bearing	-	β	Eq. 4.10
Approach Angle	-	α	Eq. 4.20
Track Angle	-	τ	Eq. 4.18
Classification	-	C_{type}	E.g. 'tugboat'
Identification Number	-	ID	Integer value

Table 4.4: Output of Navigation System.

4.6. Conclusion

The second research question: *How can perception and localisation tasks be achieved by a navigation system using stereovision?* was addressed in this Chapter. A multi-device system has been proposed to achieve mid-range perception and localisation on board inland autonomous vessels. Four depth devices onboard enable the acquisition of both textural and depth data of the surrounding environment for perception tasks, covering the directions fore, aft, port and starboard of the autonomous vessel. A tracking camera, incorporating stereo fish-eye vision sensors and an integrated inertial measurement unit is utilised to provide pose data for autonomous vessel localisation.

The multi-device perception setup enables the navigation system to maintain a constant lookout around the vessels in the collision risk regions. The utilisation a convolutional neural network to conduct object detection has highlighted its suitability to the autonomous inland vessel application. Not only does this approach provide advantages over arbitrary object detection but does so at impressive inference rates even with the limitations a small form-factor computer onboard a scale vessel. A refined approach to the acquisition of depth data avoids the unnecessary post-processing and manipulation of dense point clouds, which arguably add little value to a system centred around achieving collision avoidance.

The utilisation of a Visual Inertial Odometry approach provided a means of mid range localisation using only stereovision and IMU data. The offloading of the task to a singular compact device, allows a significant boost to small scale applications and yet the VIO approach also lends itself well to larger application setups. Secondary processing of the perception and localisation data through the mapping procedure provides a coalescent output for the Navigation System, particularly well suited to provide sufficient information to ongoing collision avoidance tasks.

Under the given computational setup, the output of the Navigation System is capable of publishing outgoing data at rate of five hertz, which is considered sufficient for the low speed application. The perception and localisation sub-tasks alone were further found to be capable of reaching frequencies closer to ten hertz.

5

Collision Avoidance for an Autonomous Inland Vessel

This chapter concerns the third research focus of this thesis, the specific question being *How can a guidance system avoid collisions with dynamic obstacles within an inland waterway environment?*. Collision avoidance is handled within the local planner of the guidance system and the sub-task division introduced in the previous Chapters of Motion Prediction, Conflict Detection and Conflict Resolution form the sectional structure within this Chapter. The contributions made by this research concern the latter two sub-tasks with the respective research sub-questions listed below being answered in Sections 5.3 and 5.4.

3a *How can collision conflict be detected between an autonomous inland vessel and a contact vessel? (Section 5.3).*

3b *How can a local path planner be configured to handle conflict resolution and avoid collision in an inland environment? (Section 5.4).*

5.1. The Guidance System

A Guidance system can be aptly divided into two main sub-tasks, a global planner and a local planner. The global planner sets a route from the departure to destination node for the AV to follow, irrespective of obstacles encountered underway. The local planner on the other hand is specifically tasked with managing temporary deviations from this global path to avoid collisions with said dynamic obstacles. The global guidance system will remain outside of the scope of this project, with attention being focused upon the collision avoidance procedure of the local planner.

Although the global planner will not form a research focus, the operation of the local planner remains reliant upon the presence of a global path therefore can be found as a module within the software structure. The global path is therefore to be specified by the user in the form of a CSV file, stating waypoints spanning from the departure to destination node and the angle of from the previous waypoint.

The local planner can be neatly divided into three sub-tasks as per the review of [24] which forms the structure of this chapter. Motion Prediction will explore the approach taken to estimate the state of dynamic obstacles in coming time steps. Conflict Detection will cover the techniques implemented to recognise collision risk within an inland environment and incite suitable action. Finally, Conflict Resolution will detail the procedure for generating the most suitable local path to avoid collision. It is worth noting that this guidance system will only be developed for, and tested with vessel-vessel interactions. Obstacles shall therefore be referred to as contact vessels (CV) within this chapter and shall apply its focus to interaction with a single vessel.

5.2. Motion Prediction

Predicting the motion of a contact vessel plays a significant role in creating a successful collision avoidance strategy. No specific contribution is made to this research field, however the implementation approach taken is detailed for context. Several methods for motion prediction were introduced during the literature study, from simple physics-based models through to more complex prediction techniques. At this initial stage of research development, the guidance system is configured with a physics based model with the implementation of more sophisticated techniques being left to form the focus of future research. Through the assumption that a contact vessel is holonomic and can move freely in the horizontal plane, the vessel's motion can be predicted over coming time steps using its current speed and track. Extending the current velocity vector to span a range future time intervals provides a prediction of future motion, albeit a crude one. Both the speed (u_{cv}) and the track angle (τ) of the contact vessel are supplied by the navigation system with their relations to the movement in the horizontal plane being defined in Equations 5.1 and 5.2 with the time step (t) being defined in seconds.

$$\dot{x}(t) = x(t) - x(t-1) = u_{cv} \cdot cos\tau \tag{5.1}$$

$$\dot{y}(t) = y(t) - y(t-1) = u_{cv} \cdot sin\tau \tag{5.2}$$

The majority of motion prediction techniques are configured for guidance systems which rely on proprietary communication such as AIS and consequently make use of the current heading of the contact vessel when predicting motion. The approach taken here will instead opt to use the current track of the contact vessel which is attainable by the current navigation system. Attempting to interpret heading in the absence of proprietary information is an convoluted task for which a reliable method is currently lacking. Although it may be argued that heading could provide a more accurate prediction of a vessel's intended course, the track more accurately represents its current trajectory.

In the coming sections, the results of motion prediction will be used to help detect conflict in sufficient time for action to be taken to avoid collision and also to help assess the collision risk of candidate local paths. Consequently, the motion prediction span (t_{pr}) must be suitably selected to ensure that sufficient warning is provided whilst not creating an overly sensitive avoidance system. The most useful value that can be supplied by motion prediction is an estimate of future position so to define the fore bound of a vessel's conflict region and assess future collision risk.

Continuing with the assumption of maintained speed and track, the velocity of the vessel and the prediction span can be used to provide the future position of the contact vessel. Whilst the speed of the vessel is variable, the prediction horizon will be a fixed value inputted by the user for each vessel type. The predicted position (x_{pr}, y_{pr}) of the contact vessel can be found by applying Eq. 5.3 and 5.4 which are derived through the manipulation of the holonomic model in Eq. 5.1 and 5.2 to account for the defined time span. The position of the vessel at the current time step is defined as ($x(t_0), y(t_0)$).

$$x_{pr} = x(t_0) + (t_{pr} \cdot u_{cv} \cdot cos\tau) \tag{5.3}$$

$$y_{pr} = y(t_0) + (t_{pr} \cdot u_{cv} \cdot sin\tau) \tag{5.4}$$

This creates a collision risk system that considers the influence of vessel speed and thus inertia upon the opportunity to avoid collision. For the model tugboat that will assume the role of the contact vessel during experimental testing, an initial prediction span value of three seconds is assigned, which at the maximum speed of $0.5m/s$ would equate to an additional contingency of 1.5 metres. It is noteworthy to mention that this definition is an intuitive estimate, this fixed value would be better defined using an expert-based method.

5.3. Conflict Detection

Conflict detection involves two major tasks, the generation of conflict regions and the monitoring of the conflict state. Conflict regions are the zones surrounding a vessel which when intersecting one another, indicate that a conflict is anticipated, or in other words, when two vessels are on a potential collision course without suitable action. An approach to generating conflict regions which are deemed appropriate for the inland waterway environment will be presented in subsection 5.3.1 and a procedure for monitoring various conflict conditions will be detailed in 5.3.2.

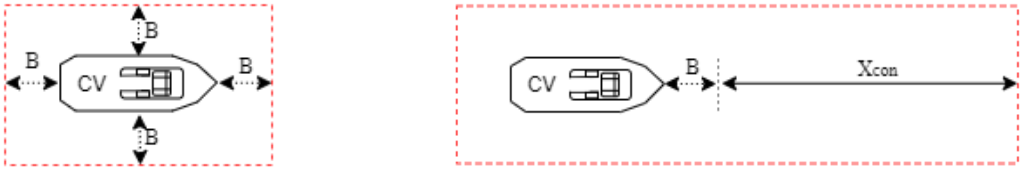
5.3.1. Conflict Regions

Conflict regions in maritime autonomy are typically defined in radial form, which whilst valid for open-water applications are less applicable to the narrow channels of inland waterways. As such, rectangular conflict regions will be proposed in this paper, inspired by the uncertainty zones presented in the Hull-2-Hull research project [30]. In the Hull-2-Hull project, the sizes of the uncertainty zones around a vessel are dependent upon the inertia of the vessel itself, which is influenced by its velocity and its mass. Whilst the mass of the autonomous vessel is known, the mass of an contact vessel in the absence of propriety communication, is not obtained by the navigation system. An estimate of the mass could be made based upon the obstacle classification, however rather than attempting to generate a pseudo-inertia value, a time based factor is applied to the obstacle velocity to yield a future position estimate as was presented in the previous section.

In this approach, each contact vessel is assigned a singular conflict region. The bound in the direction of vessel movement is influenced by the results of motion prediction where an extension is added to the base region, whilst the other base extremities remain constant and uniform. The actual size of the base conflict region is defined to be one times the beam of the vessel as indicated in Fig. 5.1a. The region remains defined this way when static, however when dynamic the region edge to the bow of the vessel will be extended by adding the contingency (X_{con}) as illustrated in Fig. 5.1b This contingency is defined by Eq. 5.5 using the motion prediction model. Although largely applicable, the definition of base region based upon vessel beam may not be well suited to large vessels operating in particularly narrow waterway stretches.

$$X_{con} = t_{pr} \cdot u_{cv} \tag{5.5}$$

The beam and length of the contact vessel are retrieved from a database which lists all vessel classifications that the neural network model can recognise and the approximate dimensions of said vessels. At this stage, the object detector is only trained to recognise the model tugboat used in testing meaning the database only has one entry, whose dimensions are exactly known. A network trained to recognise various classifications of vessel could provide a rough estimation of vessel size based upon classification.



(a) Static Base Region

(b) Dynamic Conflict Region

Figure 5.1: Contact Vessel Conflict Region. *Where B is the beam of the vessel and X_{con} is the forward bound extension.*

A different procedure is followed for the assignment of the AV conflict regions as to that used for the contact vessel. The AV requires two regions, one larger conflict region to trigger generic collision avoidance which will be referred to as the COLREG zone and one smaller region intended to incite emergency collision avoidance procedure which will be referred to as the critical zone. Fig. 5.2 illustrates how the configurations of the AC conflict regions.

The critical conflict region surrounds the port, starboard and aft edges at identical distances, defined at this stage as being one times the beam of the autonomous vessel (B_{av}). This selection also accounts for the port and starboard bow blind-spots of the AV's perception system. The size of the critical region ahead of the autonomous vessel is defined by considering its deceleration rate, to account for a worst case scenario. Meaning that after the local planner has established that no alternate route exists to circumnavigate a vessel, the AV should still

have a sufficient amount of time to either fully stop or at least significantly reduce its speed prior to collision. Although logical and applicable to some vessel types, such as the test AV used in this project, it is recognised that underactuated vessels with a higher inertia would not be so suitably catered for by this approach in its current form with extra considerations needing to be made.

The theoretical stopping distance (d) can be defined by the heuristic relation between velocity (u_{av}) and deceleration rate (a_{av}) in Eq. 5.6. Previous experimental research into the Grey Seabax vessel found that navy braking offers the best combined results considering rate of deceleration and maintenance of heading [52], with an average deceleration rate of $0.08 m/s^2$ being found. Therefore using Eq. 5.6, at the maximum velocity of $0.5 m/s$, an approximate stopping distance of $1.56 m$ is to be expected. Since the deceleration rate was determined, approximately two kilograms in navigation system hardware has been added. To account for this increase in mass and for the latency, a safety factor of 1.25 is applied to the overall stopping distance. The value F_{crit} seen in Fig. 5.2 is subsequently driven by this adjusted stopping distance as defined in Eq. 5.7, with a minimum value being set to 0.5m to account for low speeds.

$$d_{stop} = \frac{u_{av}^2}{2a_{av}} \tag{5.6}$$

$$F_{crit} = B_{av} + 1.25 \cdot d_{stop} \tag{5.7}$$

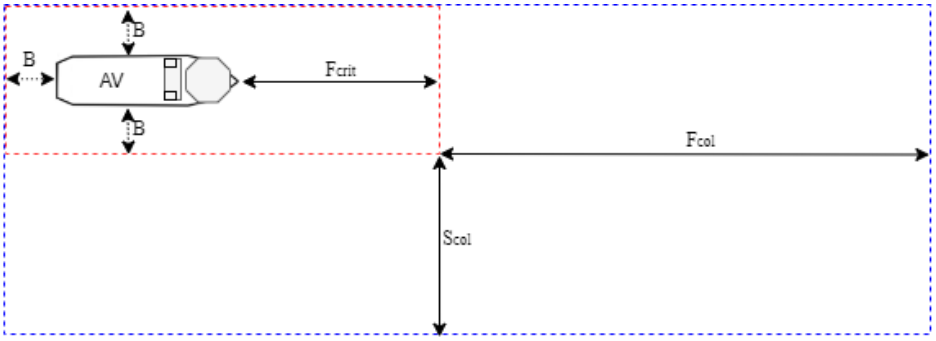


Figure 5.2: AV Conflict Regions. *Blue-dashed: COLREG region, red-dashed: critical region. F_{col} describes the fore COLREG bound, S_{col} the starboard COLREG bound and F_{crit} the fore critical bound. B refers to the beam dimension of the vessel.*

The COLREG zone surrounding the AV is not uniform as can be seen in Fig. 5.2, rather the region extremities are weighted in the fore and starboard directions to support adherence to regulation scenarios where the AV can be designated give-way status. The COLREG zone remains identical to the critical zone along the port and aft edges, due to vessel interactions on these sides yielding the AV to assume the role of the stand-on vessel. The region dimensions in the fore and starboard directions edges are driven by the vessel speed and a defined time period as defined in Eq. 5.8 and 5.9 which provide a distance in metres. For the Seabax AV, t_f is set to six seconds and t_s to three seconds with minimum dimension results being limited to 1.2 and 0.6 metres respectively.

$$F_{col} = u_{av} \cdot t_f \tag{5.8}$$

$$S_{col} = u_{av} \cdot t_s \tag{5.9}$$

5.3.2. Conflict Monitoring

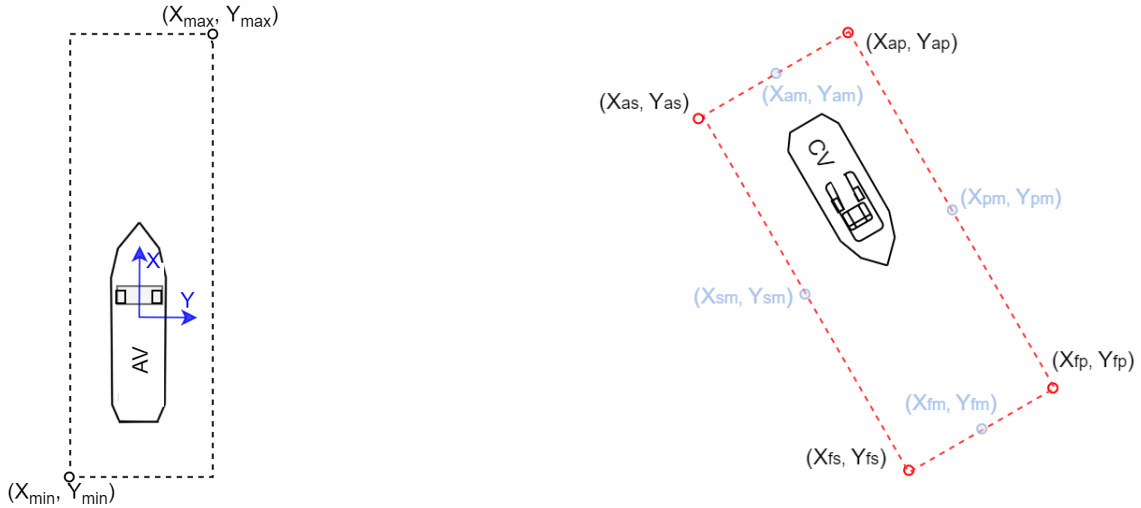
When the conflict regions of the AV and a contact vessel experience overlap, the AV will be triggered to take action to avoid a collision. The specific action taken is dependent upon which of the AV regions has detected conflict and a more detailed explanation as to the procedure will be given in the following section. This subsection will focus upon how the conditions for overlap are monitored and assessed so to send the relevant trigger to the resolution procedure.

Algorithm 3: Determination of Conflict State.

```
Result: Conflict State
if OverlapCOLREG then
    if OverlapCritical then
        conflict = 'critical'
    else
        conflict = 'COLREG'
    end
else
    conflict = 'None'
end
```

The detection of conflict involves the constant monitoring of two AV conflict regions, the COLREG zone and the critical zone. This is achieved by looping through all detected CVs and for each CV deciphering whether or not its conflict region is infringing upon one of the AV regions. In any case where overlap is detected in the the critical region, it also exists within the COLREG region. Therefore the logic in Algorithm 3 is followed.

A propositional logic approach is taken to monitor for overlap between the conflict regions of the AV and CV due to the appeal of its computational efficiency. The AV domain coordinates from the Navigation system are utilised instead of the global frame as this fixes the orientation of the AV regions, reducing problem complexity. The rectangular region of the AV can as such be separated into two ranges covering the X and Y dimensions whose bounds can be visualised in Fig. 5.3a. The orientation of the contact vessel however is not fixed, meaning that the rectangular region cannot be simply defined by two ranges. Coordinates for the CV region are instead defined as shown in Fig. 5.3b, and sub-regions are defined around these points as illustrated in Fig. 5.4. The division of elements into four regions allows for a more accurate assessment of range overlap, with both the bounded ranges and the midpoints being of interest.

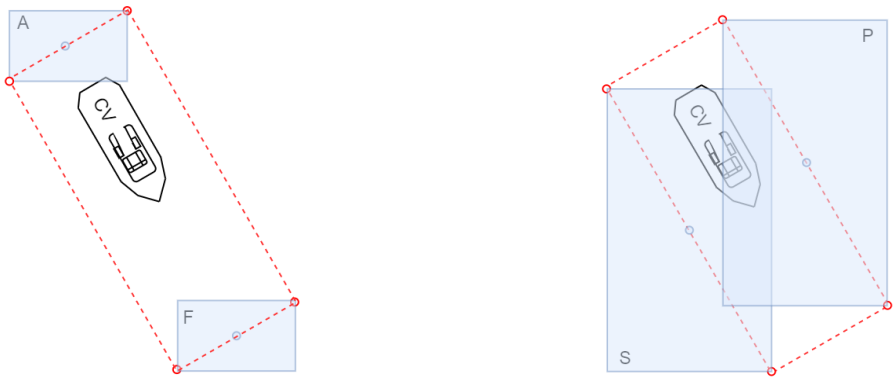


(a) Autonomous Vessel. *Black-dashed: An AV conflict region.*

(b) Contact Vessel. *Red-dashed: CV conflict region.*

Figure 5.3: Conflict Region Coordinates

The CV sub-regions are each assessed for overlap with the two AV conflict regions following the logic flow in Algorithm 3. The rectangular sub-region ranges and the midpoints are assessed independently for overlap with an AV region as per Eq. 5.10. Two propositional arguments are reviewed in each of the two coordinate dimensions (X_1, X_2, Y_1, Y_2) and when these conditions are met, a binary result is obtained indicating the presence of overlap. The terms used within these arguments refer to coordinates in Fig. 5.3. A total of eight overlap checks are made as listed in N, with the n values referring to the coordinate subscript defined in Fig. 5.3b. When these binary results are summed, the number of overlap instances (O_n) is known and utilising the cases in Eq. 5.11, the boolean conflict condition (C_o) can be assigned. Conflict is considered to be present when two or more instances of overlap are encountered.



(a) Fore(F) and Aft(A) sub-regions

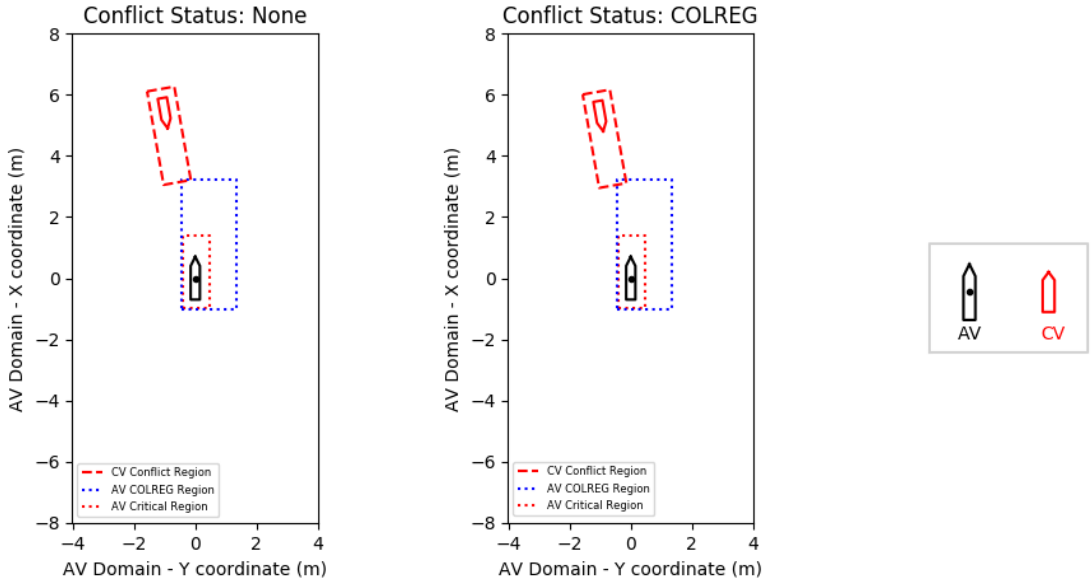
(b) Starboard(S) and Port (P) sub-regions

Figure 5.4: Contact Vessel Sub-regions. *Blue-shaded rectangles: Sub-region range, blue-shaded centroids: midpoint*

$$O_n = \sum_{n \in N} (X_1 \wedge X_2) \wedge (Y_1 \wedge Y_2) \tag{5.10}$$

$where:$
$X_1 = X_{max} \geq \text{argmin}(X_n), \quad X_2 = \text{argmax}(X_n) \geq X_{min}$
$Y_1 = Y_{max} \geq \text{argmin}(Y_n), \quad Y_2 = \text{argmax}(Y_n) \geq Y_{min}$
$N = [(fs, fp), (fs, as), (fp, ap), (as, ap), fm, am, sm, pm]$

$$C_o = \begin{cases} True, & O_n \geq 2 \\ False, & O_n < 2 \end{cases} \tag{5.11}$$



(a) Missed Detection of Conflict

(b) Successful Detection of Conflict

Figure 5.5: Conflict Detection Accuracy. *See plotted legends.*

The simple logic approach to the monitoring of conflict between the regions avoids computational expense, however the conservative approach does have limited accuracy. As the CV conflict region is only subjected to a coarse meshing of sub-regions, very small overlaps in conflict regions may be missed, which is more prevalent when a CV approaches at an angle and intersects the AV region at a corner. Figure 5.5a demonstrates such a scenario, where despite the AV and CV conflict regions marginally overlapping, conflict was not detected. This element of inaccuracy is however not deemed an issue for the application as once a slightly larger overlap is encountered as illustrated in Fig. 5.5b, conflict is successfully detected.

5.4. Conflict Resolution

A predominantly rule-based collision avoidance algorithm will be implemented in the Guidance System at this stage of development. The avoidance strategy presented focuses on single vessel interactions within an environment typical of a simple inland waterway channel. The resolution process is two-stage and is triggered when a conflict is detected as can be seen in Fig. 5.6. The subsection structure hereon will follow the two-stage resolution process, with the avoidance procedure being addressed first and the local path generation thereafter.

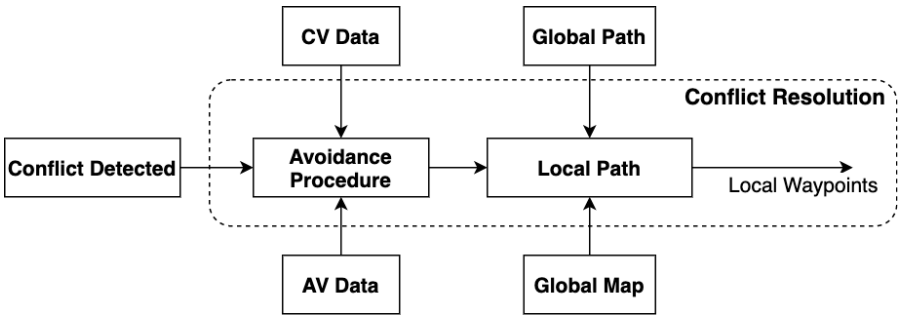


Figure 5.6: Collision Resolution Scheme

5.4.1. Avoidance Procedure

The avoidance procedure is responsible for determining the way in which the AV resolves conflict. Its purpose is to determine the status and mode of avoidance. The avoidance status details the responsibility of the AV in a given interaction and the avoidance mode identifies the type of interaction that is taking place, which indicates the specific action that should be taken by the AV to avoid a collision. The type of conflict that has been detected has an influence upon the avoidance procedure that is followed. When a COLREG conflict is detected, avoidance is conducted based upon maritime collision law and when a critical conflict is detected, an emergency procedure is followed in a final attempt to avoid collision.

5.4.1.1. COLREG Procedure

This implementation will focus on the main three COLREG interactions, head-on, overtake and crossover as were introduced in 2.4. The AV can either be allocated a stand-on or give-way status during an interaction, with a specific avoidance procedure being allocated for all give-way cases as shall be detailed further hereon. When a stand-on status is assigned to the AV, it should standby, remaining on its course as per regulations. Whilst maintaining this course, the critical region continues to be monitored by the Guidance System and in the case where the contact vessel infringes this region, the COLREG compliance can be dropped in favour of emergency avoidance.

Algorithm 4 defines the the logic procedure for determining COLREG state and an appropriate avoidance action by using the speed, bearing and approach angle. The first conditional check determines the state of the contact vessel which is necessary as a stationary vessel should be treated differently to a dynamic vessel under control. According to the COLREGs any stationary vessel can not be considered to be actively adhering to collision regulations. Consequently, when in conflict with a stationary vessel, the autonomous vessel is assigned give-way status and it must take responsibility to perform any action possible to avoid a collision. The stationary avoidance strategy consequently follows the same procedure as would be followed when overtaking a vessel.

In the case where the contact vessel is dynamic, two conditions are devised, one where the contact vessel is moving in the same direction as that of the AV, and another where the directions are opposing. In the case where the CV is moving in the same direction as the AV, three avoidance modes can be yielded. If the CV is at the

Algorithm 4: Collision Regulation Procedure

Result: COLREG Status and Avoidance

if $u_{cv} = 0$ *and not* $30 < \beta < 330$ **then**
  status = 'GiveWay'
  avoidance = 'Overtake'
else if $90 < \alpha < 270$ **then**
  if $0 < \beta < 90$ *and* $\alpha > 195$ **then**
    status = 'GiveWay'
    avoidance = 'Crossover'
  else if $\beta \geq 350$ *and* $\alpha \geq 165$ **then**
    status = 'GiveWay'
    avoidance = 'HeadOn'
  else if $\beta \leq 10$ *and* $\alpha \geq 165$ **then**
    status = 'GiveWay'
    avoidance = 'HeadOn'
  else
    status = 'StandOn'
    avoidance = 'Standby'
  end
else
  if $< \beta < 112.5$ *and* $270 < \alpha < 345$ **then**
    status = 'GiveWay'
    avoidance = 'Crossover'
  else if $u_{cv} < u_{av}$ *and not* $30 < \beta < 330$ **then**
    if $\beta > 330$ *and* $\alpha \leq 15$ **then**
      status = 'GiveWay'
      avoidance = 'Overtake'
    else if $\beta < 30$ **then**
      status = 'GiveWay'
      avoidance = 'Overtake'
    else
      status = 'StandOn'
      avoidance = 'Standby'
    end
  else
    status = 'StandOn'
    avoidance = 'Standby'
  end
end

starboard side of the AV and is approaching at a starboard angle (270° to 292.5°), the CV is considered to have a stand-on status in a crossover scenario. This therefore requires the AV to assume the give-way role and navigate behind the CV's crossover path. In the case where vessel direction is shared and the AV is approaching the stern of and travelling faster than the CV, an overtake manoeuvre needs to be conducted by the AV. There exists no hard set rule as to which side a vessel must overtake, and therefore the local planner can select a route on either side. In the case where no feasible route can be found, AV speed should be reduced until a feasible route can be found. In any other case where the AV and CV direction is the same, a stand-on state can be assumed.

When a CV is travelling in the opposite direction to the AV, three possible outcomes can be reached. The conditions for a head-on collision can be encompassed by two sets of relations between the obstacle bearing and the approach angle. This paper will take the assumption that a head on scenario can be considered in effect when a contact vessel is at a bearing $\pm 10°$ to the AV. The action that is required by a vessel in a head on scenario is that her course be altered to starboard so to cross port-port with the other vessel at a safe distance. Both vessels involved in a head-on scenario are assigned a give-way status and should both adjust course astarboard. If a vessel is within the bearing range for a head-on, it might not necessarily be on course for a head on collision, therefore the current approach angle should also be taken into consideration. If the approach angle does not suggest that the contact vessel within the head-on bearing range is on a collision course with the AV but is instead crossing its path, moving starboard to avoid this vessel may actually lead to an incident.

The second possible interaction scenario for a contact vessel travelling in the same direction is the crossover. If the contact vessel is on the starboard side and at an approach angle which clearly crosses the path of the AV, the AV should navigate behind the path of this vessel. The same contingency has been applied here to that of the head-on manoeuvre, the approach angle must be steep enough to suggest a potential crossover collision, otherwise the CV may simply be intending to pass starboard-starboard believing there is sufficient room. Again, if the AV were to take action in this case, a collision may be caused as a consequence of strictly following the COLREGs. In the case that a contact vessel is travelling in the same direction as the AV and does not yield one of these two potential outcomes, the AV can assume a stand-on status and standby for any potential emergency avoidance that may prove necessary.

5.4.1.2. Emergency Procedure

When a contact vessel appears to not be adhering to the collision regulations or the current avoidance action does not sufficiently reduce risk, an emergency procedure is required to protect the AV from collision where possible. Little can be done by the AV to avoid a collision to the sides or stern of the vessel, as under normal operation it has no meaningful control over these directions. However if a collision is anticipated ahead of the vessel and the COLREGs cannot be followed, emergency avoidance should be conducted in an attempt protect the AV.

Algorithm 5: Emergency Avoidance Procedure

Result: Emergency Avoidance
status = 'Emergency' **if** $u_{cv} = 0$ **then**
    if *not* $5 < \beta < 355$ **then**
        avoidance = 'EmergencyStop'
    else if $\beta > 180$ **then**
        avoidance = 'HardStarboard'
    else
        avoidance = 'HardPort'
    end
else if $\beta > 315$ **then**
    if $\alpha < 165$ **then**
        avoidance = 'HardPort'
    else
        avoidance = 'HardStarboard'
    end
else if $\beta < 45$ **then**
    if $\alpha > 195$ **then**
        avoidance = 'HardStarboard'
    else
        avoidance = 'HardPort'
    end
else
    avoidance = 'CollisionImminent'
end

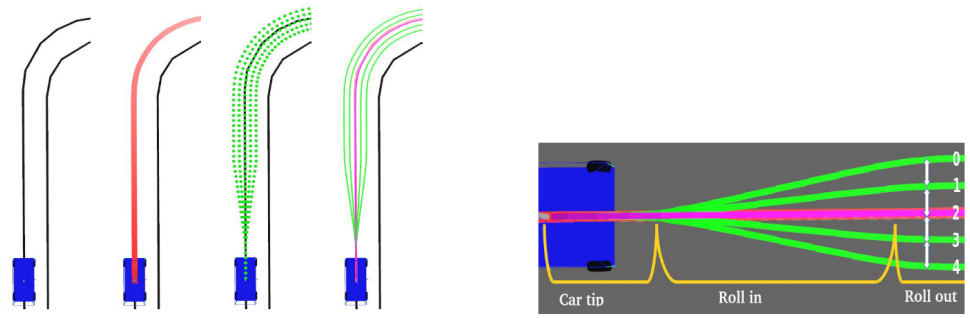
When an infringement to the critical region is detected, the emergency procedure is triggered. As was the case for the COLREG procedure, the direction of the avoidance trajectory shall be decided based upon the bearing and approach angle of the contact vessel, see Algorithm 5. If a stationary vessel is dead ahead within the critical region, the safest option for minimising risk would be to perform an emergency stop as an avoidance direction is difficult to call in the absence of the approach angle. Hard Port and Hard Starboard avoidance modes encourage the AV to conduct a sudden and significant change in direction to avoid a collision. In any case where the contact vessel is at an approach angle crossing the AV path, emergency avoidance should move to the opposite direction to forgo a manoeuvre that would turn the AV towards future danger. When a contact vessel is located to one side of the AV and not crossing its path, the avoidance should favour moving in the opposing direction of the contact vessel's position. When the conflict region is infringed in any other scenario, there is little that the AV can do to avoid incident and so a collision can be considered highly likely.

5.4.2. Local Path Planner

As was discussed in Chapter 2, not all of the existing maritime solutions to collision avoidance are suitable for application on the inland waterway. The architecture of the inland waterway perhaps bears a closer resemblance to road networks, albeit lacking the set lane divisions. As such, this guidance system utilises an adapted implementation of the Roll-out Trajectory Generation (RTG) technique developed by Darweesh et al. [37] for autonomous cars and made available as part of the Autoware platform. Within this section, the scheme followed for the generation of candidate paths using RTG shall be presented and the procedure for path selection under CORLEG and Emergency avoidance modes shall be discussed.

5.4.2.1. Roll-out Trajectory Generation

In line with the primary aim of this project to develop a standardised programming structure, the adapted technique used in this thesis has been implemented in Python as opposed to its native C++ and included within the Guidance ROS package. Whilst the fundamental procedure behind this new implementation remains similar, a few major changes have been made during the generation procedure.



(a) Rollout Trajectory Generation

(b) Trajectory Smoothing

Figure 5.7: Autoware Local Path Planner [37].

Firstly and perhaps most importantly, the roll-out generator has been optimised based upon the maritime collision regulations. Rather than generating roll out trajectories to both sides of the vessel (see Fig. 5.7a), the approach has been enhanced to only generate trajectories in the direction which the avoidance mode requires. For the majority of COLREG avoidance cases, trajectories are only generated astarboard. Only in the case of an overtake manoeuvre where the contact vessel lies to the starboard side of the global path shall trajectories be generated aport. In the case of the emergency avoidance, the AV can be required to turn hard starboard or hard port so to avoid an obstacle that poses high risk. Therefore one roll-out trajectory is generated in emergency scenarios which immediately creates a path as far away as possible to the port or starboard side to increase the chance of avoidance and to minimise latency.

Additionally, whilst the Autoware implementation makes use of a car tip region and velocity dependent roll-in trajectories to smooth transition (see Fig. 5.7b), the local planner in this project simplifies trajectory generation somewhat. The inclusion of such features has been neglected in this implementation due to lower speeds in the maritime environment and in particular with the current RAS test vessels. Furthermore, within rule 8 of the collision regulations, it is stated that any alteration to course and/or speed should be significant enough to be readily apparent. Therefore any path smoothing that were to be conducted would need to ensure that this requirement not be infringed, a problem that is not encountered with discretised steps. An element of smoothing is nonetheless provided by the path-following based trajectory tracker, with the acceptance radius surrounding a target waypoint allowing for smoother transitions between trajectory angles. The only other contingency made for smoothing is for target parallel trajectories that are in close proximity to the waterway edge.

The generation of candidate trajectories works simply by comparing the current position of the AV and the navigable edge of the waterway channel to define a bound and then creating trajectories that run parallel to the current path, with spacing between one times the width of the AV. The number of trajectories that can be generated to the side of the AV is defined in Eq. 5.12, where d is the distance to the edge of the waterway from the current path and B_{av} is the vessel beam. As can be seen, the vessel beam is further deducted from the distance to ensure all trajectories have sufficient clearance from the waterway edge and the number of trajectories that could fit within this adjusted distance is calculated by dividing by the vessel beam and flooring the result to the nearest integer.

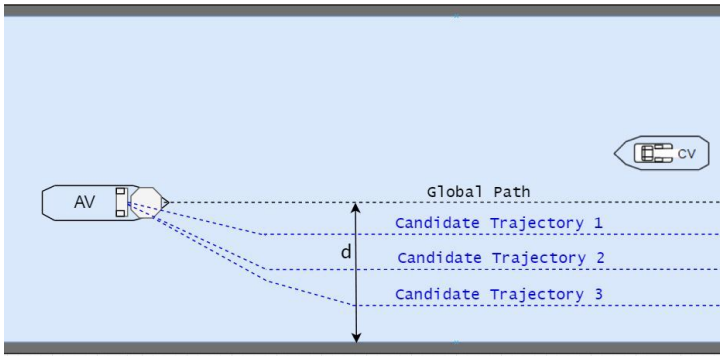


Figure 5.8: Example of Roll-out Trajectories. B_{av} *is the vessel beam, d is the distance to the bank.*

$$\lfloor n \rfloor = \frac{d - B_{av}}{B_{av}} \tag{5.12}$$

A visual example of candidate trajectory generation can be seen in Fig. 5.8. Each trajectory is generated at a parallel distance of one times the vessel beam from the global trajectory or the superseding candidate trajectory. Smoothing the roll-out trajectory itself is generally neglected as a clear indication of manoeuvre intention is required by regulations and waypoint progress incorporates an acceptance radius which in itself offers an element of path smoothing. That being said, waypoint generation does require an element of path smoothing when the generated trajectory is close to the the edge of the waterway. Candidate trajectory three in Fig. 5.8 can be seen to have a smoothed roll-out region so to ensure the vessel has adequate opportunity to change its course to meet the parallel path.

5.4.2.2. Path Selection

The path selection procedure determines the most suitable route from the candidate trajectories and is only required for the COLREG protocol as in an emergency, only one trajectory is generated. To select the best candidate trajectory in a given interaction, integer weights are assigned to each path and a simple optimisation problem is solved to minimise cost. Three influential factors were identified for the inland application and these regard the distance efficiency of a path, the expected avoidance action based upon CORLEGs and the risk of collision that a path carries.

The distance weight (d_i) is added to each of the trajectories based on the distance from the current path in an order ascending magnitude with distance as shown in Table. 5.1. The COLREG weight (c_i) is assigned based upon the specific avoidance mode which indicates the magnitude of expected avoidance, i.e. a head-on or overtake scenario would yield lower weights closer to the current path whilst a crossover which requires more significant action would lead to higher weights closer to the current path.

Candidate Trajectory	**Distance Weight (d_i)**	**COLREG Weight (c_i)**		
		Head-on	**Overtake**	**Crossover**
1	1	1	2	3
2	2	2	1	2
3	3	3	2	0

Table 5.1: Integer Distance and COLREG Weights

The final weight aims to discourage paths that are close to the contact vessel's predicted motion as these potentially increase collision risk. Consequently, the proximity of a candidate trajectory (P_x) to the CV's predicted position in the y-axis reference plane is taken as the quantification of this risk and is defined in Eq. 5.13. This predicted position (y_{sp}) is retrieved from the motion prediction module and y_{ct} is the parallel y-axis reference of the candidate trajectory. A negative proximity value suggests that the CV is predicted to cross the candidate trajectory in a direction not expected by collision regulations and thus yields a heavy weight penalty as collision risk is high. For positive proximity values, a linear trend is followed whereby the larger the proximity value the

lower the weight that a candidate trajectory carries. The assignment uses a factor of the AV's beam to generate the integer weights as shown in table .

$$P_x = y_{ct} - y_{pr} \tag{5.13}$$

Proximity (P_x)	**Collision Risk Weight** (r_i)
$P_x < 0$	10
$0 < P_x < B_{av}$	5
$B_{av} \leq P_x \leq 3B_{av}$	1
$P_x > 3B_{av}$	0

Table 5.2: Integer CV Proximity Weight. *Where B_{av} is the Autonomous Vessel Beam and P_x is the proximity.*

The optimal candidate trajectory is then selected by solving the simple optimisation problem defined in Eq. 5.14. This solution simply summates the three trajectory weights into one trajectory cost and selects the candidate trajectory with this lowest total cost. In any case where the optimisation problem provides two trajectories which carry the same cost, decision favours in the direction of caution, selecting the trajectory with the further distance to increase opportunity for avoidance. The weighting system whilst suitable at a developmental level would again benefit from an expert-based method to determine the optimal weights as the current definition has limitations and oversights.

$$T_w = min \sum_{i=1}^{n} (d_i + c_i + r_i) \tag{5.14}$$

The autonomous vessel follows the local path until the avoidance action is considered sufficient and it is safe to return to the global path and continue the journey. Once the CV has been tracked past a bearing associated with the aft side of the vessel, namely ($150 < \beta < 210$), avoidance can be considered complete.

5.5. Guidance Output

The Guidance System provides global and local paths for the autonomous vessel to follow, as well as tracking the progress along these paths. Within the GNC structure, the output of the Guidance system is the current target waypoint which is published to the ROS network as a string message at a rate of five hertz. The format of this message is a one dimensional array with two entries, detailing the X Y coordinates of the waypoint. The target waypoint is primarily intended as an input for the high-level controller to generate a target heading for the AV as detailed in Chapter 3.2.5.

5.6. Conclusion

The third research focus resulted in a Guidance System being developed which incorporates a tailored collision avoidance protocol for the challenges of the inland waterway. A new configuration for conflict detection regions has been proposed to meet the challenges brought on by the architecture of the inland waterways and the nature of the interactions that take place on it. The techniques sees the use of rectangular vessel domain regions whose dimensions are influenced by a function of the vessel speed. The autonomous vehicle is assigned two conflict regions, one that when infringed indicates a scenario whereby collision regulations can be followed and a second critical region where emergency avoidance procedure should be followed.

The collision resolution approach sees the integration of collision regulations and a secondary emergency procedure within a rule-based implementation compatible with the data available from a stereo-vision based navigation system. This two tier system prevents the AV from blindly following the collision regulations when a contact vessel is recognised to be non-compliant or an extremely close encounter requires additional action to be taken. The local path planner incorporates a roll-out-trajectory generation technique that has been adapted and optimised for application on an inland autonomous vessel. The planner operates by generating candidate trajectories based upon the determined collision avoidance mode and the limitations of the navigable waterway channel. The optimal path is then selected from the candidate trajectories by applying three weight variants based upon the added distance, collision regulations and collision risk and solving the subsequent optimisation problem.

6

Results

System evaluation is critical in understanding the performance of the sub-systems, which leads to the final sub-question. The guidance sub-system can be verified using simulations, however the navigation sub-system requires experimental testing in order to assess its performance. Furthermore, inline with the main question it is crucial to understand the level of collision avoidance that can be achieved using a stereovision based navigation system. The components of the fifth and final research question: *How well do the guidance and navigation systems perform under experimental testing and what are their limitations?* shall be covered in this chapter. The specific sub-questions this chapter covers are detailed below.

5a *How does the stereovision based perception system fair under precision/recall testing and position error KPIs?*

5b *How well does the stereovision based localisation perform under the KPIs of position and heading error?*

5c *Can the developed guidance and navigation systems enable the autonomous avoidance of collisions under various staged scenarios?*

6.1. Experimental Setup

To conduct experimental testing, both a test environment and experimental hardware are required. The Grey Seabax that assumes the role of the autonomous vessel during testing has already been introduced in Chapter 3, however the contact vessel that is used during testing is yet to be formally introduced. The Tito Neri vessel that shall assume the role of contact vessel and shall be introduced in the following subsection 6.1.1. The test tank where the experiments are conducted and the tracking setup used to provide ground truth pose data on the vessels within this environment shall be discussed in the remainder of this section.

6.1.1. Contact Vessel

The Tito Neri (Fig. 6.1) is a 1:30 scale model tugboat measuring at one metre in length with a beam of 0.3 metres. During experimental testing the Tito Neri can be controlled by means of a joystick, allowing for adaptive scenarios to be met on demand. The vessel is actuated by two azimuth thrusters located at the stern of the vessel and a bow thruster. The Tito Neri vessel was used to acquire data during the training of the neural network for obstacle detection and thus falls under the classification of 'tugboat'. The experimental testing conducted in this research project will not test multiple-vessel interactions meaning that the Tito Neri is the only contact vessel present at this stage of development.

Figure 6.1: Tito Neri Model Tugboat

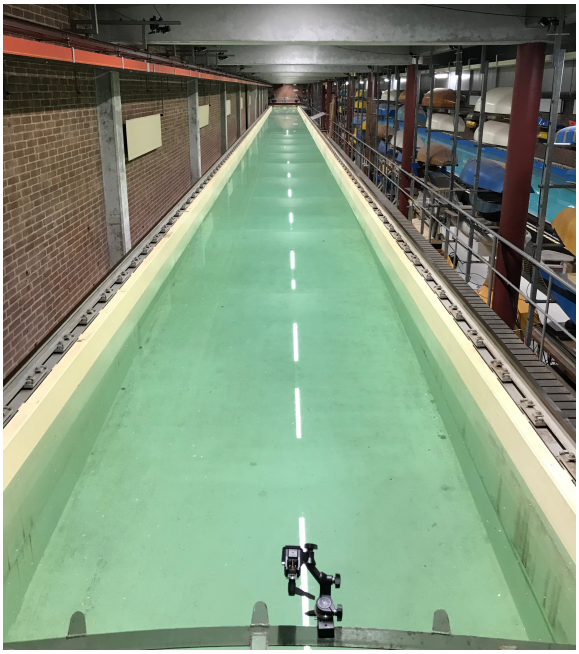


Figure 6.2: MTT Towing Tank

6.1.2. Test Tank

As the application focus for the autonomous vessel is the inland waterway network, it is most suitable to select a test environment that mimics the restricted channel widths that are synonymous with the inland waterway. Within the MTT department at TU Delft, one of the towing tanks provides a suitable an environment for this stage of testing. The towing tank utilised is 85 metres in length with a width of 2.75 metres and can be seen in Fig. 6.2 above. The narrow channel width of the tank emulates the inland network as desired, however the entire length of the tank will not be required for testing, leading to the workspace for this project being reduced down to a twenty metre stretch of the tank, with the full width still being utilised.

Although the test tank provides a suitable replication of inland waterway architecture, the hydrodynamic conditions that a vessel would be subjected to are not replicated. External influences could be applied to the test environment using lab equipment however as the research scope only aims to provide the guidance and navigation systems with experimental proof of concept. The capacity of the control system to deal with external forces is left outside of the scope of this research. Consequently experimental testing shall take place in still waterway conditions. Artificial lighting is present within the test environment that would not be present in a real-world application, however it is noted that this stage of research does not aim to review performance in extreme light conditions so the fixed nature of the lighting provides consistency between testing occasions.

6.1.3. OptiTrack 3D Tracking Setup

The towing tank is further equipped with an OptiTrack 3D tracking system comprised of twenty-eight cameras covering a forty metre span of the tank, which has been calibrated to sub millimetre accuracy. The test vessels being used within the experimental environment are equipped with passive markers so to be identified by the tracking system as rigid-bodies. The pose data of the vessels acquired from this fixed tracking system will be accepted as the ground truth during the calculation of the navigation evaluation metrics.

The collection of experimental data from the OptiTrack setup is achieved by streaming rigid-body pose from the MotiveTracker software the ROS network via a client server protocol. The relevant pose data can then be subscribed to by any node within the network, with the further ability to record the published messages to a ROS bag file for post-experimental analysis. The use of the ROS network for recording this data provides timestamped messages, which proves invaluable when aligning data sets. As the Navigation system developed in this project is also built within a ROS environment, data forms are further consistent between the ground truth values and the experimental measurements.

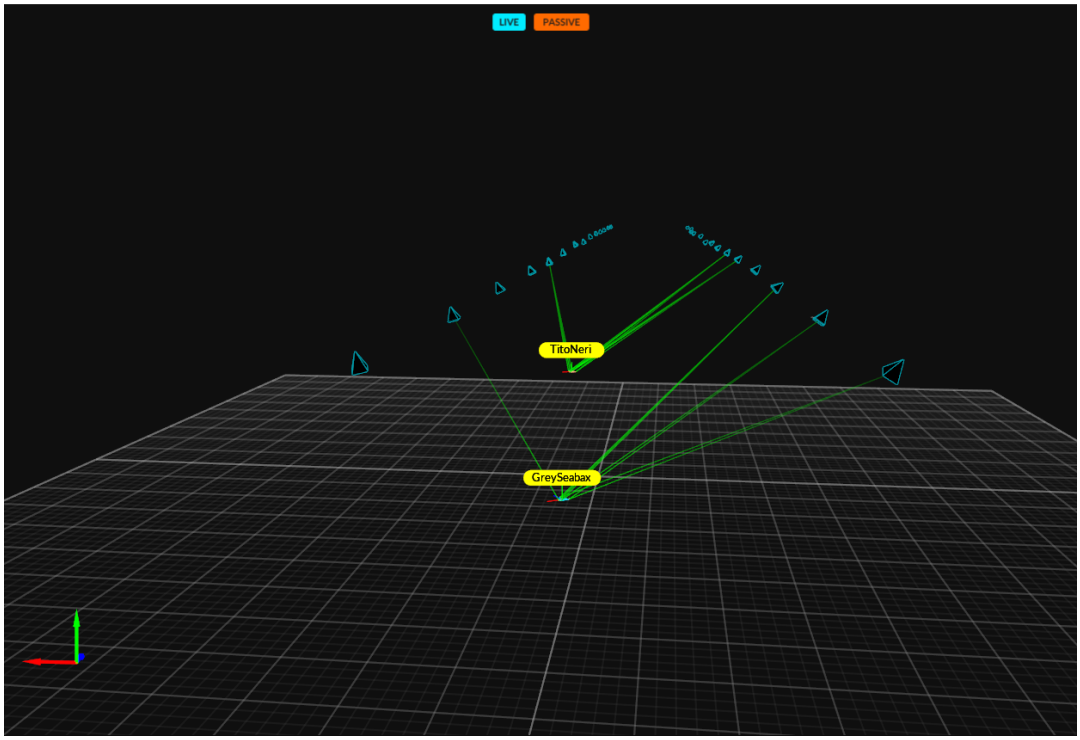


Figure 6.3: OptiTrack 3D Tracking System

6.2. Navigation System

In the literature review, a number of key performance indicators were identified for their suitability in reviewing perception and localisation system performance. The perception KPIs quantify performance of the object detection and obstacle localisation procedures, as divulged upon in subsections 6.2.1 and 6.2.2 respectively. Pose data acquired by the localisation module is evaluated in subsections 6.2.3 where both position and orientation data is reviewed.

6.2.1. Detection Precision and Recall

Testing the performance of the object detection procedure in a maritime environment is important to understand the specific ability of the chosen technique for the application. Testing is conducted using multiple sets of onboard footage that were recorded within the test tank environment. As experimental testing is only setup for interaction with the tugboat variant, the following evaluation is conducted using this singular object type.

Arguably the most important feature of an object detector for application in an autonomous navigation system is robustness. An entirely robust model would detect all obstacles that are present within a frame. The metrics that can be used to best assess this robustness are recall and precision, the KPIs which were introduced in Chapter 2. A good object detector should maintain a high precision as recall increases.

Recall can simply be considered to quantify the ability of the detector to find all of the relevant cases within an environment. To achieve 100% recall, the detector must be capable of identifying every obstacle that exists within a data-set. On its own, recall presents a limited metric for assessing robustness due to the fact that it does not account for cases where irrelevant detections are made. The Precision metric can be used to fill this gap and assess the relevance of each detection within an accuracy threshold. More specifically, what is the percentage of correct positive detections within the allowable threshold. This is of course influenced by the occurrence of false positives and the occurrence of true positives whose bounding boxes lie outside of the permissible threshold. The Intersection-over-union (IoU) threshold for this application has been set to 75% as opposed to the 50% or 95% alternatives. The perception system does not require a perfectly defined bounding box for its functionality meaning some contingency can be permitted for the detection precision. However accuracy should still be sufficient to ensure the correct obstacle depth data can be extracted from the ROI. Consequently the middle-ground in this case of threshold selection represents the most suitable choice.

Evaluating under the IOU threshold of 0.75, the obstacle detector was found to achieve a total precision of 97.4% with a recall of 94.6%. This means that out of all the true positives that existed within the testing dataset, 94.6% were detected by the model and of these successful detections, 97.4% were within the acceptable precision threshold. The relation between precision and recall can be further assessed by plotting the curve of the two metrics. Figure 6.4 shows the tail end of the precision recall curve obtained through the evaluation procedure. The model achieved 100% precision rating up to a recall of 93.6%, where the precision then can be seen to gradually reduce. Within the 5.4% of recall that the model failed to achieve, 2.5% were false negatives and 2.9% were false positives.

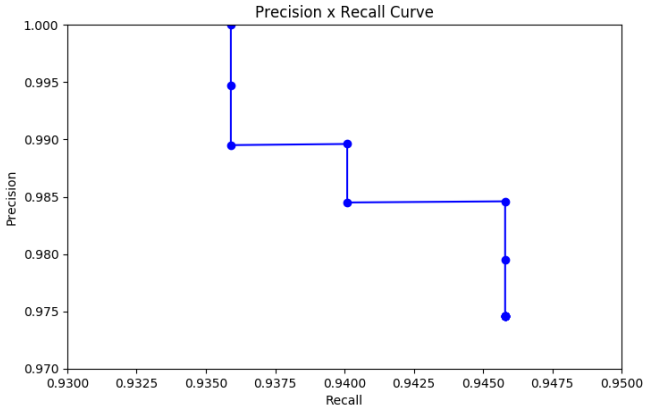


Figure 6.4: Precision Recall Curve above 94% Recall.

Within the tested environment, the precision remained high as the recall increased which indicates a robust detection model. Whilst highlighting high performance for the current detection model, there are some limitations to this evaluation. Firstly, the neural network was trained and tested within the same environment meaning that this evaluation only holds for these conditions. Furthermore, as only one obstacle was present in the tank, there was little opportunity for other objects in the environment to be detected as false positives. When the level of research development demands it, this evaluation procedure should be repeated in multiple environments with more obstacles to gain a more transferable metric.

6.2.2. Perception Position Error

The second key performance indicator used to quantify the performance of the perception system is the root mean square position error as was defined in Eq. 2.3. This key performance indicator provides a good evaluation of the quality of depth data that can be obtained by the stereovision devices utilised by the perception system as well as the technique used to extract the closest obstacle point. The perception system extracts an obstacle position within the device frame which has been used to generate this metric as further manipulations of this data would cause the metric to deviate away from its focus on perception.

The evaluation was conducted using the pose data of the two vessels provided by the OptiTrack setup described in section 6.1. By aligning the passive markers associated with each vessel with the stereovision device origin and the COP of the contact vessel, the absolute distance between the vessels within the 2D reference frame can be obtained. This value is taken as the ground truth value and can be compared with the euclidean distance between the depth camera position and the position the perception system associates with the COP.

Data was gathered within the ten metre range covered by the perception system, with a total of 280 data points being gathered across three tests with varied vessel positioning. The perception system is limited to ten metres as this is the operational range of the stereovision device, which was considered during its selection and operating outside of this zone would cause significant reductions in data quality. The resultant position error recorded at each of these data points can be seen in Fig. 6.5, with the individual error measurements being indicated by the black scatter plots on the graph and the root mean square error of the dataset being indicated by the red line. The root mean square position error can be quantified from this evaluation as 0.31 metres.

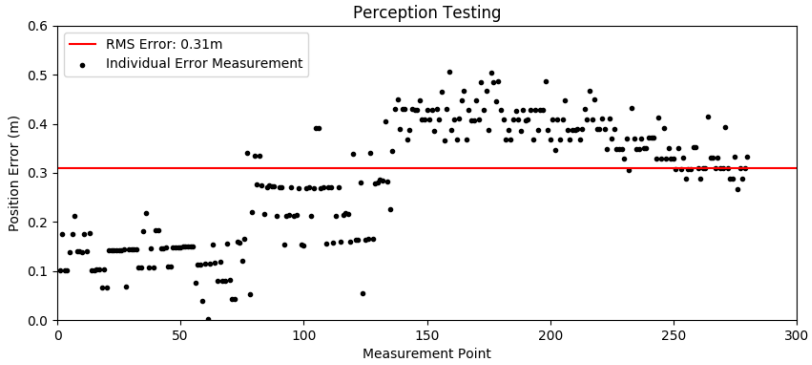


Figure 6.5: Perception System Position Error

Taking the application into account, the root mean square position error that has been obtained from experimental testing bears a magnitude similar to that of the model scale vessel beams. Such an error in open water applications would not be so troubling, however the close interactions on the inland waterway lead this error to have a far greater significance. In the following chapter means of reducing this error shall be discussed.

6.2.3. Localisation Position and Heading Error

The performance of the localisation system was evaluated using tracking data acquired across multiple experimental tests. The two key performance indicators that have been selected for the evaluation are position error (ATE_{pos}) and heading error (ATE_{rot}), which are both defined by the root mean square deviance between the acquired sensor value and the accepted ground truth value. The mathematical definitions of these two metrics were introduced in Eq. 2.1 and 2.2.

Experimental testing to evaluate localisation performance is conducted using two path types, a slalom/zigzag style path and a straight path. As an inland vessel will generally move along straight trajectories, it is beneficial to review the performance of the localisation procedure under typical sailing conditions. When a vessel traverses along a trajectory that includes sharp turns and sudden manoeuvres, the conditions within which a localisation system must operate are somewhat difference. As this research project proposes the use of stereovision for collision avoidance, the performance achievable on a slalom style path is of particularly significant interest. As test tank length being used during the experiments is restricted to a twenty metre length, the paths that shall be generated this space shall not be exceed a fifteen metre length. Consequently the evaluation of the localisation procedure shall focus upon this range of the waterway.

Testing the performance of the localisation system to track movement along a straight path yielded promising results. Figure 6.6 provides a data plot from a straight path evaluation, with the black line indicating the ground truth, the blue line indicating the sensor reading and the red area indicating the position error. The localisation system can be seen to match the ground truth trajectory very closely along the majority of the path length, with the only error appearing towards the end of the path, where the position acquired by the localisation system is seen to lag slightly behind the ground truth position.

Experimental localisation testing along a slalom path yielded, as anticipated, slightly less accurate position data. Figure 6.7 shows position data acquired when the vessel traversed a slalom path, with the plot legend being identical to the straight path. It can be visually observed that when the vessel is following a more dynamic trajectory that the localisation performance begins to dwindle, an important consideration for a vessel using localisation data for collision avoidance procedures. The error can be seen to increase in magnitude in response to manoeuvres in the lateral direction and as was the case with the straight path, the error at the end of the path highlights that the localisation system lags behind the ground truth.

Both the straight paths and slalom paths exhibited a correlation between the magnitude of error and the path length, a behaviour that is commonly faced by localisation systems that utilise a Visual-SLAM approach. This behaviour is frequently characterised by the sensor drift whereby position error accumulates over the length of a path. As well as being visually evident in the path plots, this behaviour can be further investigated by analysing the data along divided sections of the path. In this case, the data shall be divided into two sections, halfway down the waterway length within which the path runs.

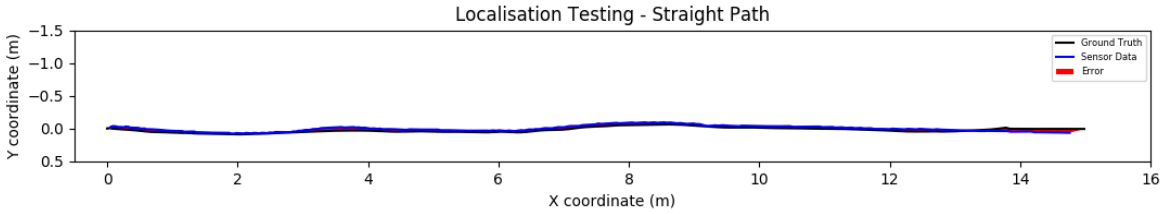


Figure 6.6: Localisation System Testing - Straight Path

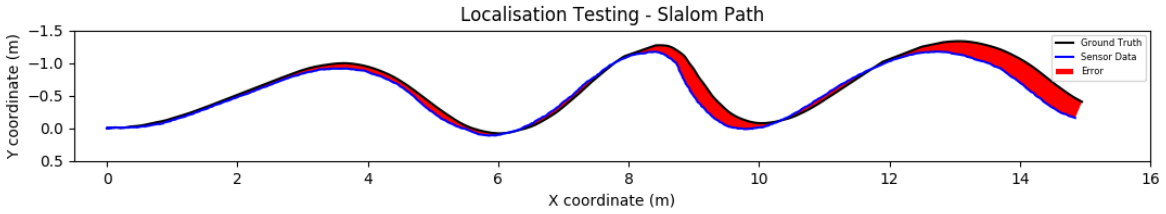
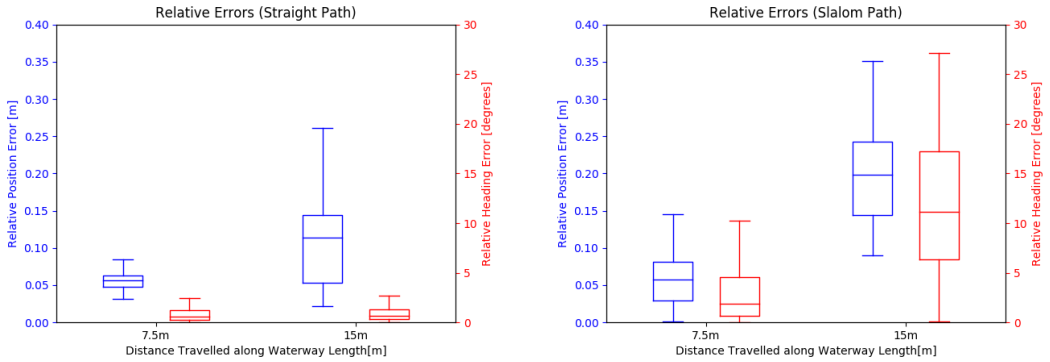


Figure 6.7: Localisation System Testing - Slalom Path

The relative position and rotation errors between the ground truth and the sensor readings are taken at given time steps along the path. As the localisation system publishes data at ten hertz, this time step is taken as 100 milliseconds. Boxplots can be used to provide a representation of these relative errors along the path length for analysis. A boxplot diagram can be found for both path variants in Figures 6.8a and 6.8b, where the position and heading errors share the same plot area, but are associated with different y-axis, as distinguished by the colour coded plots and axes. The boxplots themselves are comprised of a number of features that help paint a clearer picture of the position and heading errors. The central box of each plot indicates the interquartile range of the errors which is further split horizontally by the median line. The whiskers highlight the upper and lower quartiles, which are bounded by the caps.



(a) Straight Path

(b) Slalom Path

Figure 6.8: Boxplot representation of relative position and heading errors for divided sub-trajectory lengths. Boxplot colour corresponds to y-axis of the same colour. Specifically. blue: position error and red: heading error.

Reviewing the box-plots it can be clearly seen that the conclusion that was made from the visual path error was not unfounded. The relative position error can be seen to increase significantly in the second half of the path length division, both in the straight and slalom paths. The relative heading error can be seen to remain fairly consistent between the sections of the straight path, however this trend is not shared by the slalom path, with a significant increase in heading error being observed in the latter path section.

With a graphical evaluation of the localisation and position errors having been presented, a numerical metric can be applied by quantifying performance based upon the root mean square deviations of the position and heading. The quantification of localisation system performance is taken as the average errors gathered across the full length of both path variants. The position error metric can consequently be defined as 0.129m and the heading

Waterway Length	KPI	Slalom Paths	Straight Paths	Average
0 - 7.5 (metres)	ATE_{pos}	0.076m	0.059m	0.068m
	ATE_{rot}	4.24°	1.04°	2.64°
7.5 - 15 (metres)	ATE_{pos}	0.216m	0.125m	0.171m
	ATE_{rot}	13.6°	1.14°	7.37°
0 - 15 (metres)	ATE_{pos}	0.153m	0.105m	0.129m
	ATE_{rot}	9.49°	1.11°	5.30°

Table 6.1: Localisation Key Performance Indicators over varied path lengths and types

error as 5.3°. Due to the discussed variations in localisation performance based upon the trajectory shape and the length of the path, a break down of the key performance indicators achieved in different scenarios is provided in Table 6.1. This breakdown is particularly useful when one is deciding upon a dependable operational range for the sensor type.

The evaluation of the localisation system was conducted using data gathered under typical operational behaviour, however there were a number of instances during experimental testing where the reliability of the localisation system came to be questioned. On occasion the localisation system fails to provide sufficient data, either failing to recognise when the vessel departed from its initial starting position or suddenly jumping to an incorrect location on the map. This behaviour can again be attributed with the "kidnapped robot" problem of which localisation systems suffer. This issue occurs when the tracking camera essentially gets lost within its environment, or incorrectly associates its position with a similar one encountered at a different point. Whilst infrequent, the occurrence of this issue has significant implications for ongoing processes, highlighting the necessity for data fusion with another additional sensor set.

6.3. Guidance System

Verification and validation of the implemented collision avoidance procedure within the Guidance System is conducted using simulation. During experimental testing, the Guidance System performance dependent upon the quality of information received from the navigation system and upon the ability of the control system to follow the provided path. As such, primary evaluation of the guidance system using controlled inputs alone is important. In this section the conflict detection and resolution procedures are tested across a host of scenarios to verify the results obtained are that expected. For the majority of the evaluations, the simulation space is equivalent in dimensions to the tank used for experimental testing, allowing for assessment within the target environment. Sufficient performance in these simulation tests therefore enables the fourth question to be answered by ensuring the guidance procedure is correctly implemented prior to experimental testing.

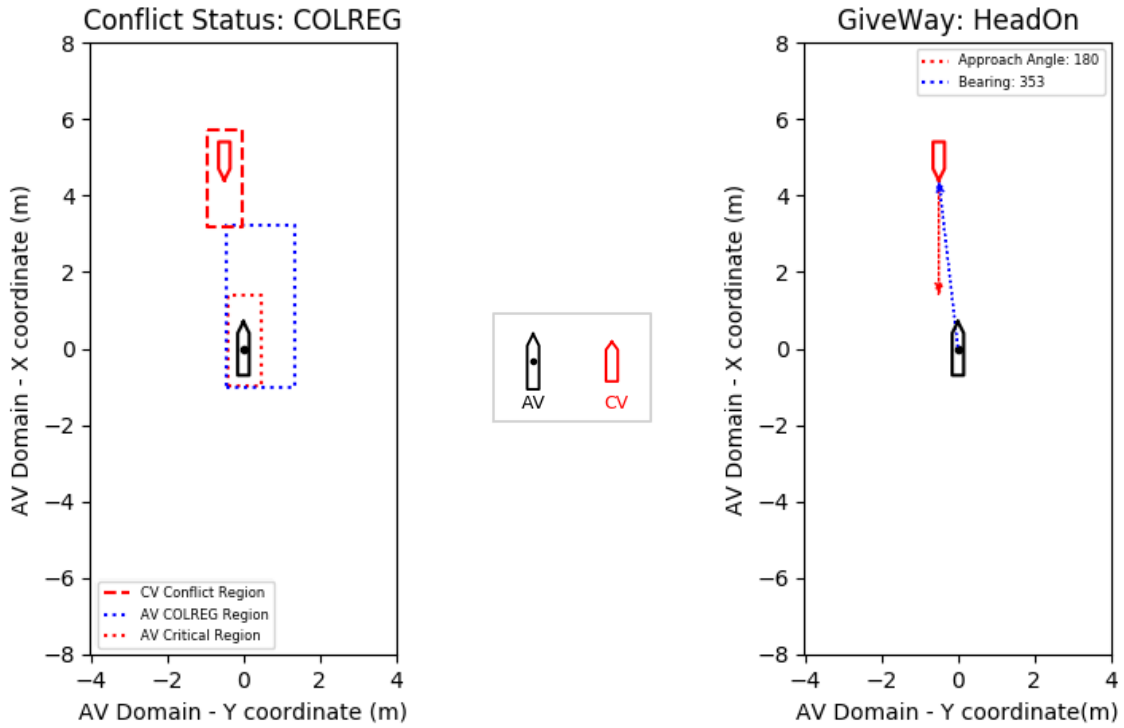
Simulation testing shall consider two scenario sets. One set of scenarios within which the autonomous vessel should adhere firmly to the collision regulations and a second where an emergency collision avoidance strategy is required for the vessel to avoid collision. Within each simulation, three main evaluation points exist. Firstly, the ability to detect conflict in a scenario is assessed through the graphical analysis of the simulation plot. Secondly, the ability of the collision resolution procedure to correctly identify the current collision scenario and determine the AVs status of responsibility and a mode for avoidance. This is assessed graphically and numerical from the simulation plot. Finally, the resultant local path that is generated by the guidance system to avoid a collision is assessed for both its suitability to avoid the collision and its optimality.

6.3.1. COLREG Avoidance

Three COLREG interaction types have been implemented within the collision avoidance strategy and conditions do not always call for particular action. To verify the behaviour of the avoidance system in a host of scenarios, each of the three COLREG interactions have been subjected to ten simulated scenarios. The vessel parameters differ in each of these scenarios to assess interaction performance across different conditions. The key parameters are the bearing, approach angle and the vessel speeds. The results of the simulation are assessed to determine whether or not the guidance system generates an action compliant with the regulation for the given interaction. It is further assumed in all of these scenarios that the contact vessel is compliant with the regulations. Within this section a few of the most noteworthy simulation results shall be presented and discussed with a full recording of results being displayed in Appendix B.

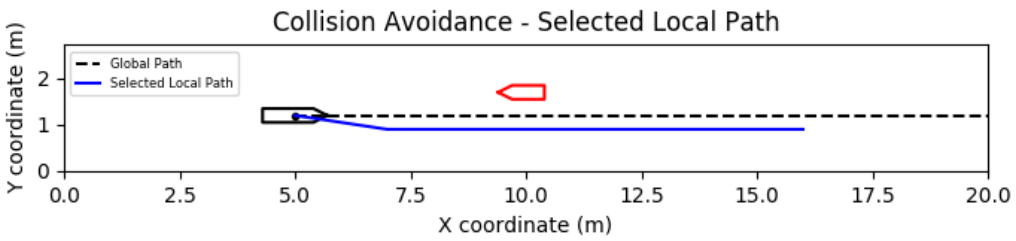
6.3.1.1. Head On

As both vessels involved in a head-on scenario are required to take action to avoid a collision, only a slight deviation from global path to the starboard side is required under normal circumstances. This behaviour can be demonstrated by the guidance system as indicated by the results in Fig. 6.9. Should a contact vessel be slightly to the starboard side, yet a head-on scenario is still identified, the guidance system should account for this by selecting a local path further starboard as demonstrated in Fig. 6.10. Alternatively, in the case that the AV is already at the starboard extremity of the waterway when a head-on collision is detected, it cannot move to its starboard side. So despite being recognised as a head-on scenario, the AV should remain on its course allowing the contact vessel to avoid the collision as shown in Fig. 6.11. If the CV fails to do this in time, emergency avoidance will be triggered. The implemented collision avoidance procedure can be seen to succeed in all of these scenarios to perform in a manner compliant with collision regulation.



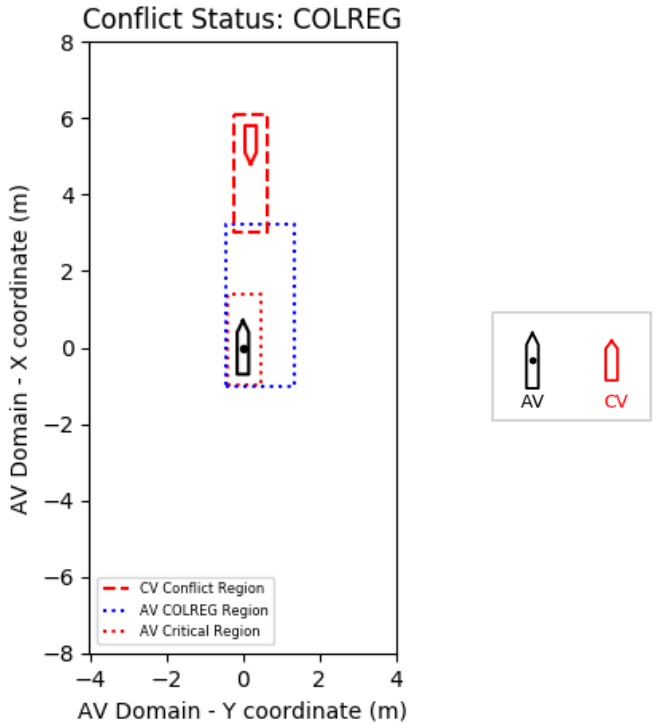
(a) Conflict Detection

(b) Avoidance Mode

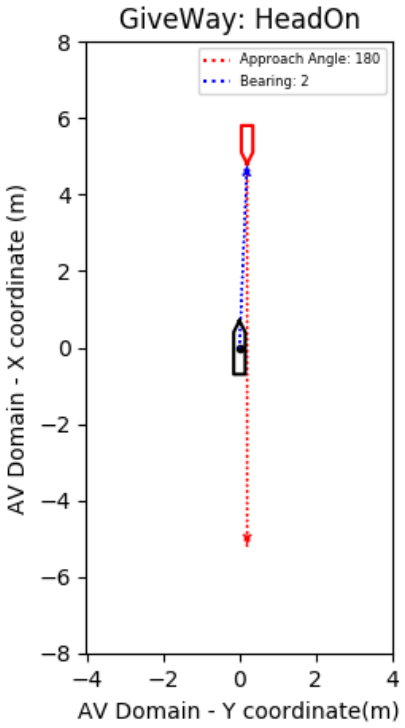


(c) Collision Resolution - Local Path

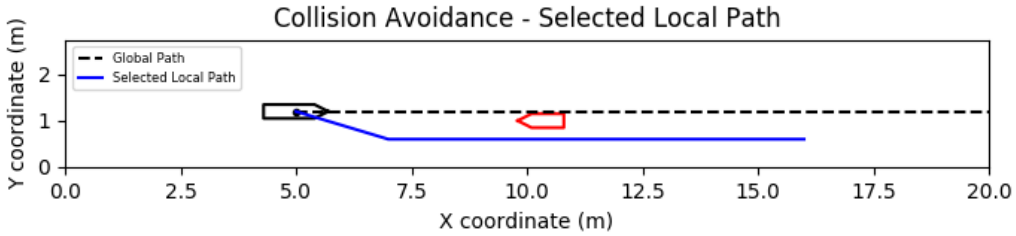
Figure 6.9: Typical Head-On Simulation Result. *Refer to plotted legends.*



(a) Conflict Detection

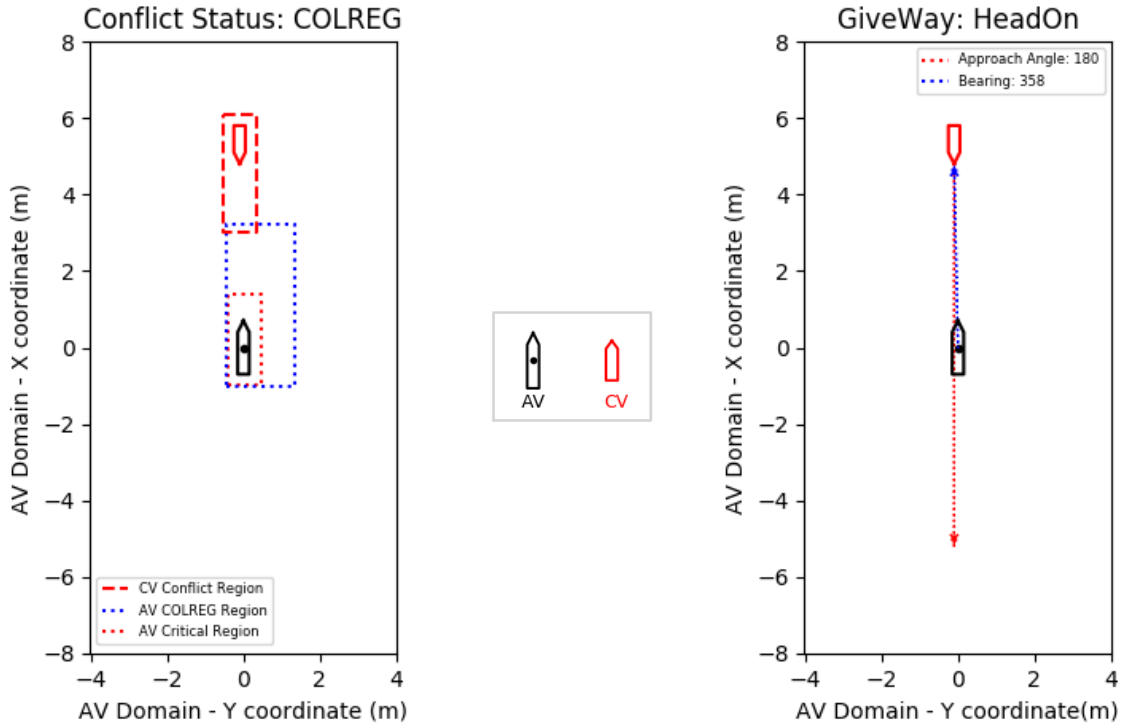


(b) Avoidance Mode



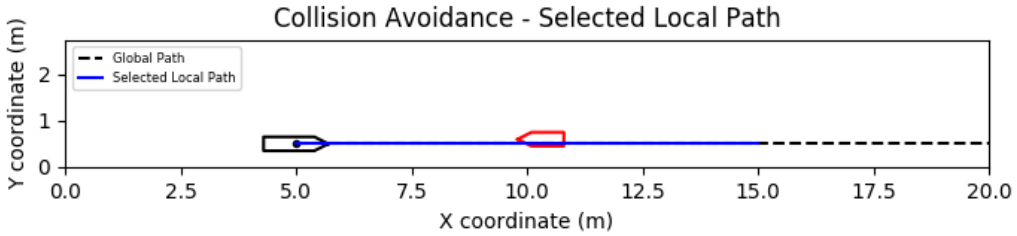
(c) Collision Resolution - Local Path

Figure 6.10: Starboard Head-On Simulation Result. *Refer to plotted legends.*



(a) Conflict Detection

(b) Avoidance Mode

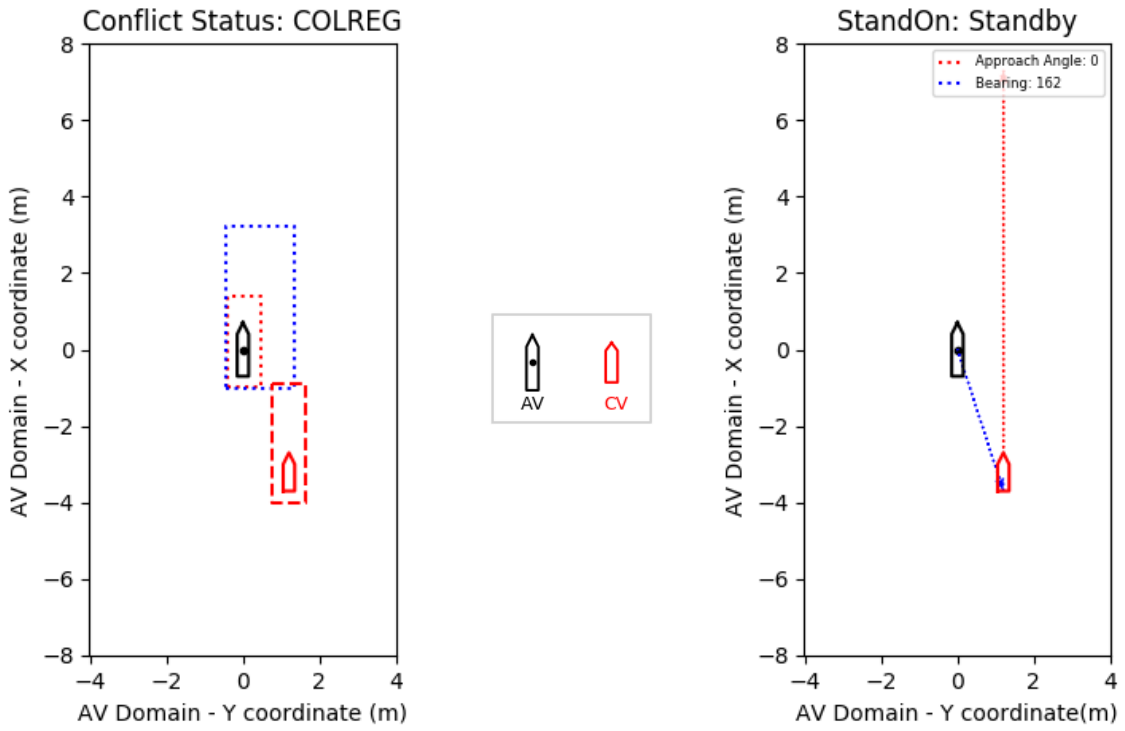


(c) Collision Resolution - Local Path

Figure 6.11: Head-On Simulation Result at Waterway Edge. *Refer to plotted legends.*

6.3.1.2. Overtake

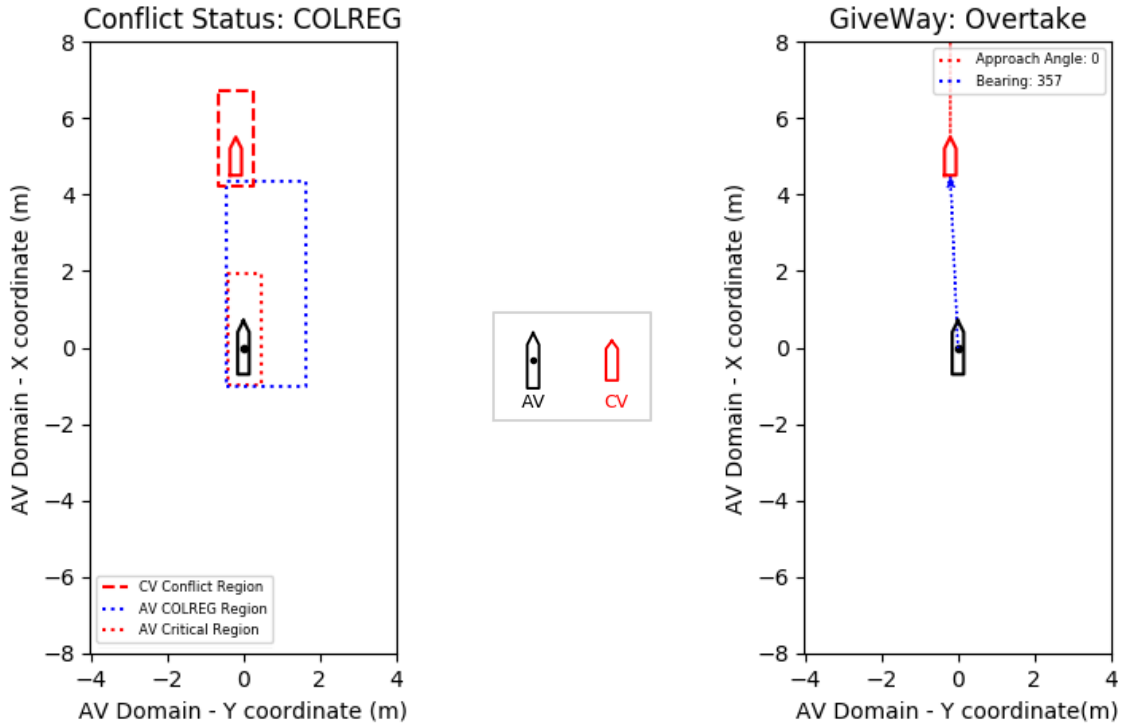
Overtaking is the second of the avoidance protocols that is up for evaluation. When approaching a contact vessel travelling in the same direction and slower than the AV, an overtake manoeuvre should be triggered. Unlike with a head on scenario, during an overtake manoeuvre the autonomous vessel needs to take full responsibility in avoiding a collision as only one vessel is assigned Give Way status. There exists no hard set rule regarding which direction an overtake manoeuvre should be conducted in, however the implementation saw the starboard side be preferable where a feasible path exists. Fig. 6.13 represents an overtake scenario where conflict is correctly identified and a new path is assigned to the starboard side. Fig. 6.14 on the other hand shows another overtake scenario, however in this case the manoeuvre is correctly conducted to the port side as it is recognised that no feasible path exists to the starboard side. An alternative outcome of a simulated overtake scenario is one where the AV takes the role of the stand-on vessel as seen in Fig. 6.12.



(a) Conflict Detection

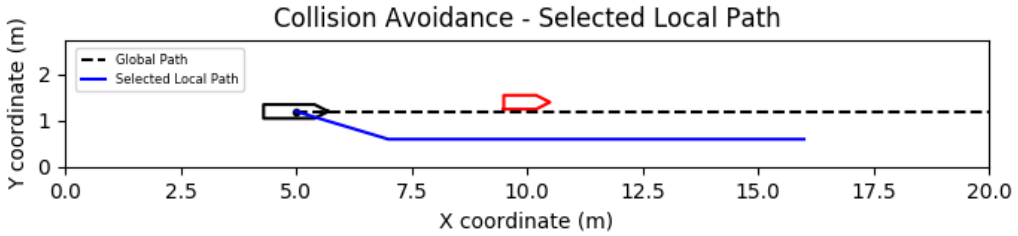
(b) Avoidance Mode

Figure 6.12: Stand On Overtake Simulation Result. *Refer to plotted legends.*



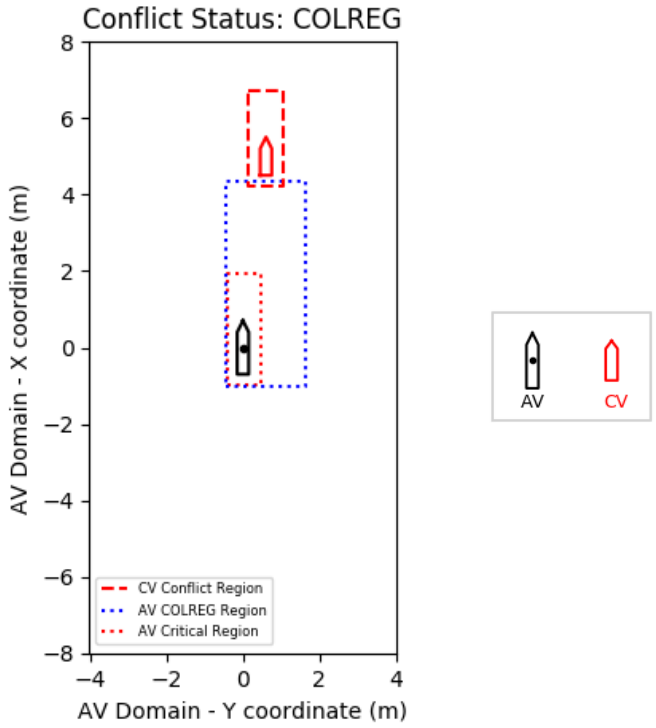
(a) Conflict Detection

(b) Avoidance Status and Mode

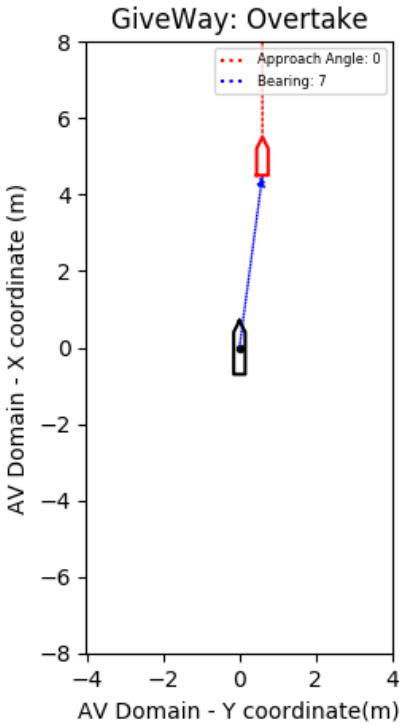


(c) Collision Resolution - Local Path

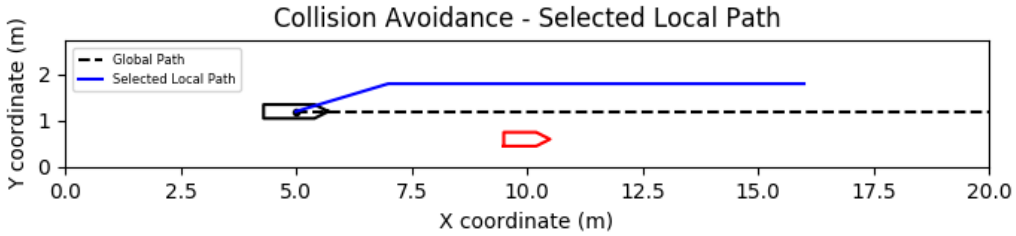
Figure 6.13: Starboard Overtake Simulation Result. *Refer to plotted legends.*



(a) Conflict Detection



(b) Avoidance Mode. *Refer to plotted legends.*

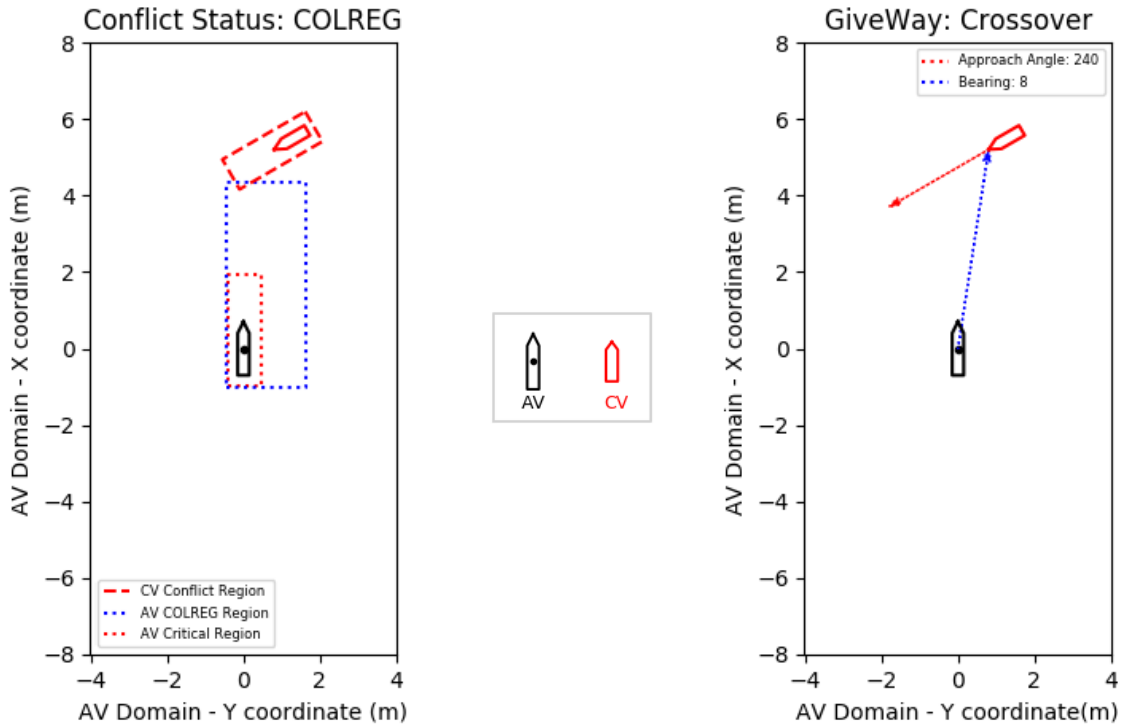


(c) Collision Resolution - Local Path. *Refer to plotted legends.*

Figure 6.14: Port Overtake Simulation Result. *Refer to plotted legends.*

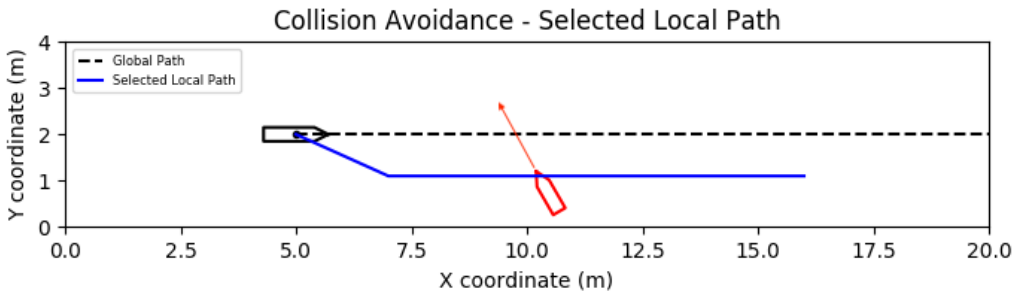
6.3.1.3. Crossover

Whilst head-on and overtake scenarios are particularly likely to occur on the inland waterway, crossovers are far less likely. Nonetheless in some scenarios such avoidance may well be triggered and an inland autonomous vessel must be able to handle such a scenario as per regulations. In order to trigger a crossover response, the simulation workspace had to be widened from the dimensions of the towing tank as this width was insufficient for evaluation. In the crossover scenario, the AV may be assigned either the give way or stand on status. In the former, the AV should change its course to navigate behind the path of the crossing contact vessel as shown in the simulation result of Fig. 6.15. It is worth noting that in Fig. 6.15c the predicted motion of the CV in the coming time steps is indicated by the red arrow which can be seen to cross the path of the AV. Therefore the local path selected allows the AV to navigate behind the vessel as per regulations. Furthermore, the procedure has proven it can successfully determine when a stand on crossover scenario is valid as indicated by the results ins Fig. 6.16.



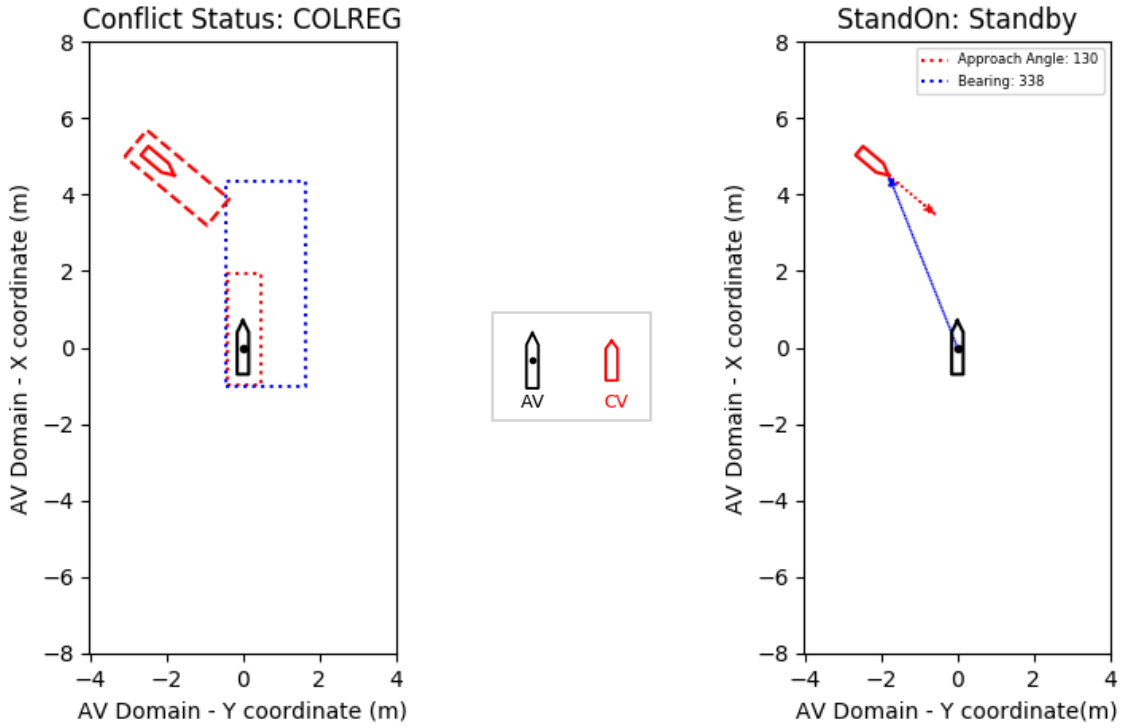
(a) Conflict Detection

(b) Avoidance Mode



(c) Collision Resolution - Local Path. *Red arrow indicates predicted motion of the CV.*

Figure 6.15: Give Way Crossover Simulation Result. *Refer to plotted legends.*



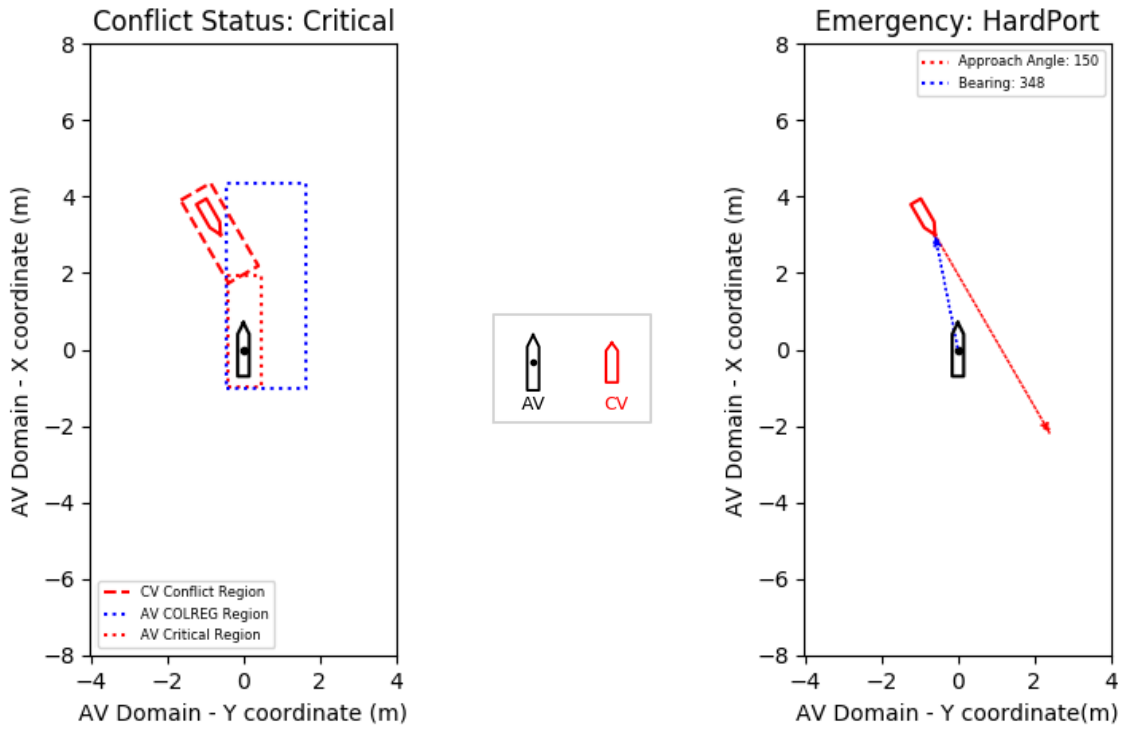
(a) Conflict Detection

(b) Avoidance Mode

Figure 6.16: Stand On Crossover Simulation Result. *Refer to plotted legends.*

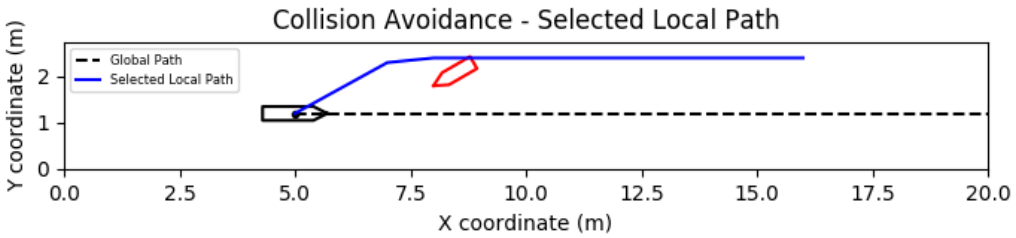
6.3.2. Emergency Avoidance

A second set of scenarios that need to be evaluated under simulation were those that required emergency avoidance procedure. This mode may be triggered when the action taken during COLREG avoidance was not significant enough to avoid risk or when the contact vessel is not compliant with regulations. Two changes to the local path can be expected from the AV in the emergency case. A hard starboard manoeuvre is initiated when the port direction is deemed to have a higher likelihood and vice versa for the hard port manoeuvre. Fig. 6.17 shows a scenario where despite the contact vessel being on the port side, the hard port manoeuvre is initiated to avoid collision. The reason for this situation developing was due to the contact vessel not assuming its give way status within the COLREG crossover scenario. The contact vessel is recognised to still be crossing the path of the AV and so to avoid collision the AV would be best to change course to the direction in which the contact vessel is not moving. In Fig. 6.18 a similar scenario is encountered, only this time the cause of the incident was that the contact vessel failed to take action in a head on scenario. As can be seen however, the vessel is not crossing the path of the AV and therefore a hard starboard avoidance mode is initiated to prevent collision.



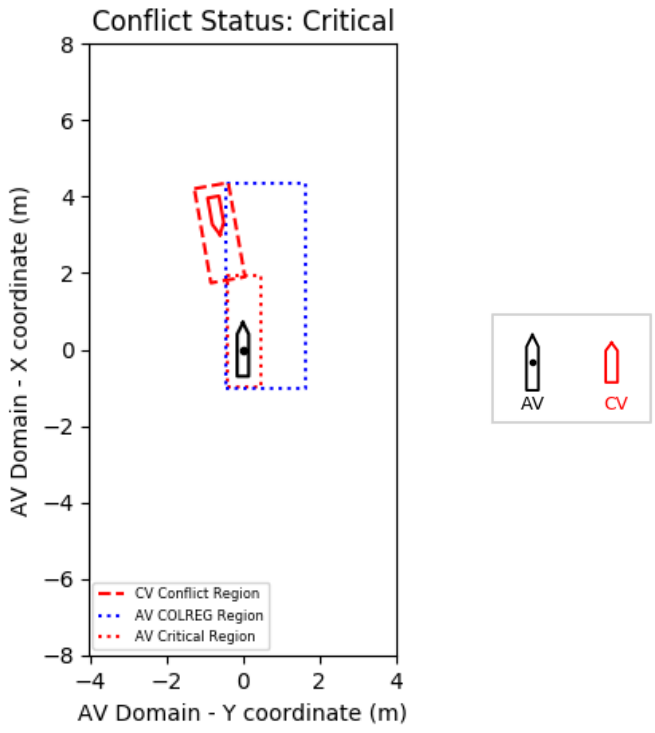
(a) Conflict Detection

(b) Avoidance Mode

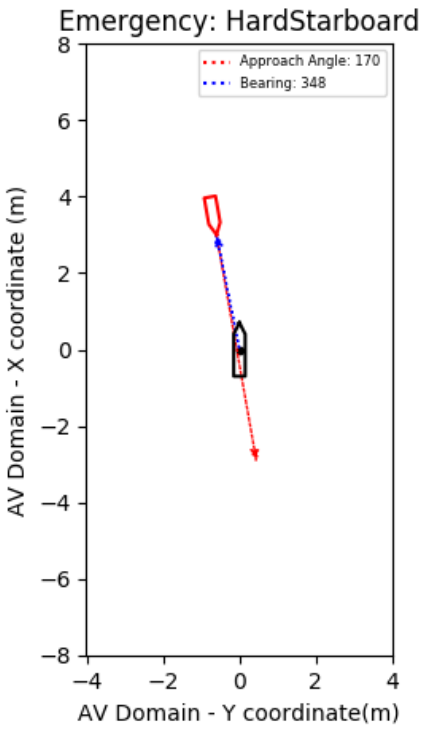


(c) Collision Resolution - Local Path

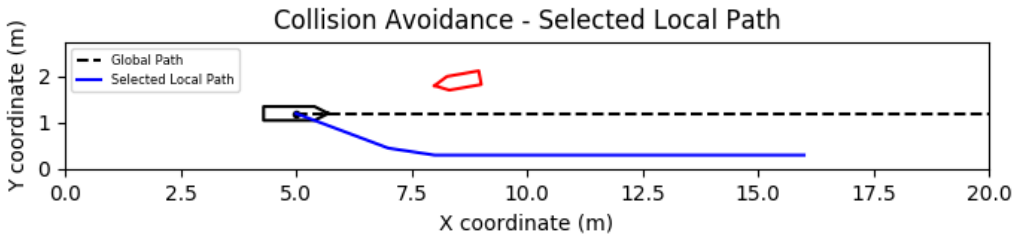
Figure 6.17: Hard Port Simulation Result. *Refer to plotted legends.*



(a) Conflict Detection



(b) Avoidance Mode



(c) Collision Resolution - Local Path

Figure 6.18: Hard Starboard Simulation Result. *Refer to plotted legends.*

6.4. Experimental Collision Avoidance

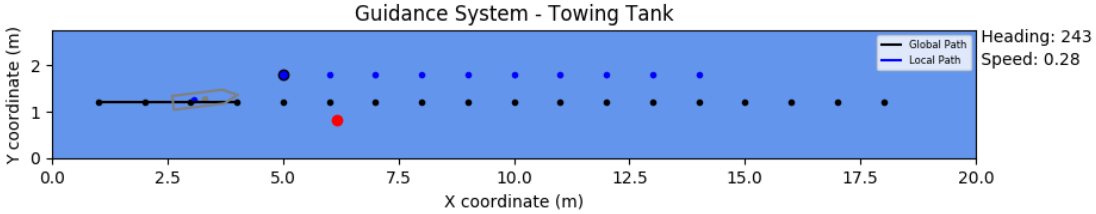
Experiments were conducted under three interaction scenarios, static, head-on and overtake. Due to the restricted tank size in relation to the vessel size, it proved very difficult to emulate a crossover scenario within the experimental testing tank. Consequently, the crossover scenario has not been experimentally tested with these experiments being left for future research in less restricted waters. Instead the ability of the system to recognise and avoid static obstacles has been assessed, which is also of importance and a requirement of the collision regulations, specifically rule 18.

The static type scenario involves a contact vessel which is recognised to not be under active control and stationary within the global environment, meaning that the autonomous vessel must assume full responsibility to avoid a collision. The head-on scenario sees the autonomous and contact vessel travelling along the waterway section towards one another and in general scenarios, both vessels should be seen to take action to avoid a collision by passing port-to-port. In the case where the initial change of course is not significant enough, the autonomous vessel should recognise this and take action to further alter its course and avoid collision. The overtake scenario regards the case where the autonomous vessel and contact vessel are travelling in the same direction. When the autonomous vessel approaches a contact vessel which is travelling slower, it should change its course to safely overtake. Again, in the case where an initial course is not considered sufficient to avoid collision, the local path should be changed again to avoid collision.

Ten experiments were conducted per avoidance mode so to evaluate the consistency of performance and define the experimental collision avoidance KPIs. During the collision regulation focused testing, it has been assumed that the contact vessel is also complying to the regulations. Therefore any emergency avoidance scenarios that are triggered are done so due to the fact that the original action was found to be not significant enough.

6.4.1. Static

When conflict is detected with a static obstacle in the environment, the collision avoidance procedure instructs the autonomous vessel to conduct an overtake manoeuvre. The local path considered most suitable shall thereafter be selected to either the port or starboard side depending upon the specific scenario. An example set of results from one of the experimental tests can be seen in Figure 6.2. In sub-figure (a) the AV can be seen to recognise the static contact vessel (red point) and generate a path around the vessel to avoid a collision. In sub-figure (b) the guidance log can displays the procedural results from the successful encounter. Having detected COLREG conflict, the Guidance system went on to correctly recognise the vessel as being static and assigned the overtake avoidance mode.



(a) Experimental Scenario and the Selected Local Path for Avoidance

```
--------------------------Guidance Update--------------------------
[INFO] [1604385221.010231]: COLREG conflict was detected
[INFO] [1604385221.011535]: AV has been assigned GiveWay status.
[INFO] [1604385221.012271]: Static Vessel Detected.
[INFO] [1604385221.012845]: Overtake avoidance mode activated.
[INFO] [1604385221.013765]: New local path has been assigned.
[INFO] [1604385221.014193]: Changing course to avoid collision.
```

(b) Procedural Response to the scenario from the Guidance System

Figure 6.19: Example Results from Static Experiment

Test Number	Collision Avoidance			Path Optimality		
	Compliant	Semi-compliant	Failed	Added Distance	Collision Vicinity	Metric
1		✓		0.55	0.60	0.92
2		✓		0.58	0.58	1.00
3	✓			0.24	1.20	0.2
4	✓			0.23	1.02	0.23
5	✓			0.28	1.36	0.21
6		✓		0.43	0.98	0.44
7			✓	-	-	-
8		✓		0.69	1.14	0.61
9	✓			0.20	0.94	0.21
10		✓		0.39	1.31	0.30
Average	40%	50 %	10%	0.40	1.01	0.40

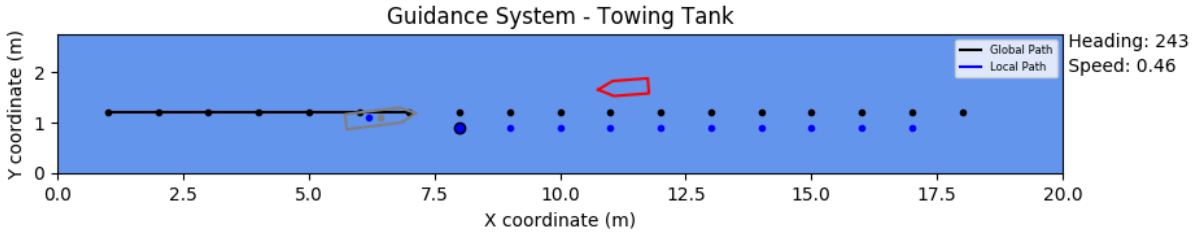
Table 6.2: Experimental Results from Static Scenario. *Compliant: successful avoidance as per procedure, semi-compliant: successful avoidance however not exactly per procedure, Failed: unsuccessful avoidance. Metric: path optimality key performance indicator (Eq. 2.7).*

Table 6.2 presents the numerical results for all ten static scenarios and Fig. C.1 in the Appendix provides the graphical result for each test. The table is comprised of all the KPIs for each test as well as the totalled results for the static tests in the last row. In 90% of the scenarios the guidance and navigation systems enabled the AV to autonomously avoid collision, however only 40% were fully compliant with the regulations and one was a total failure. The path optimality metric came out at 0.40 which is relatively low, meaning that the static avoidance procedure yields reasonable performance results regarding the collision risk and efficiency.

The total failure in test number seven was caused by a localisation system failure through which the the AV and CV objects were not mapped correctly and collision was not avoided. In tests 1 and 10, only semi-compliant avoidance was experienced due to an incorrect scenario being recognised. Instead of recognising the static vessel and initiating an overtake manoeuvre, the system recognised a head on scenario. Whilst this still lead to successful avoidance, it was not fully compliant with the regulations. The reason behind this false recognition was due to incorrect navigation data most likely due to an accumulation of the defined localisation and perception errors. Of the other three tests that yielded semi-compliance, the emergency scenario was triggered too soon.

6.4.2. Head On

The head on scenario in Fig. 6.21 provides one of the outcomes of the ten experimental tests with the rest being presented in Fig. C.2. It can be seen that the system correctly detects the head on scenario to hand and generates a regulation compliant path. Table 6.3 below presents the results of all of the ten tests and it can be seen that only 30% of all scenarios recorded fully compliant avoidance, with 20% failing to avoid the collision entirely. As well as a drop in avoidance success over the static case, the path optimality was also seen to worsen with a metric of 0.57 being obtained.



(a) Experimental Scenario and the Selected Local Path for Avoidance

```
--------------------------Guidance Update--------------------------
[INFO] [1604384534.566991]: COLREG conflict was detected
[INFO] [1604384534.567933]: AV has been assigned GiveWay status.
[INFO] [1604384534.568622]: HeadOn avoidance mode activated.
[INFO] [1604384534.569581]: New local path has been assigned.
[INFO] [1604384534.570215]: Changing course to avoid collision.
```

(b) Procedural Response to the scenario from the Guidance System

Figure 6.20: Example Results from Head On Experiment

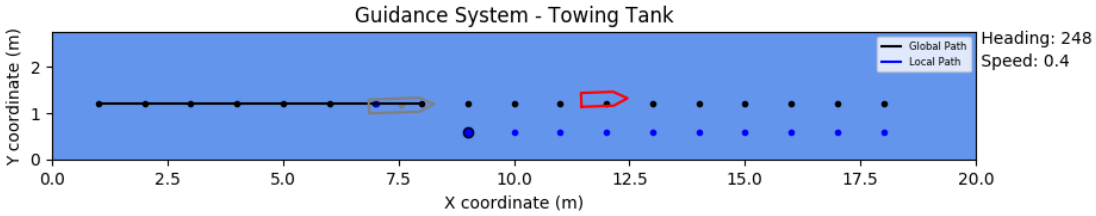
Test Number	Collision Avoidance			Path Optimality		
	Compliant	Semi-compliant	Failed	Added Distance	Collision Vicinity	Metric
1	✓			0.38	0.76	0.5
2		✓		0.69	0.94	0.73
3		✓		0.50	0.62	0.8
4			✓	-	-	-
5	✓			0.44	1.02	0.43
6		✓		0.4	0.5	0.8
7			✓	-	-	-
8	✓			0.2	0.8	0.25
9		✓		0.54	1.21	0.45
10		✓		0.46	0.6	0.77
Average	30%	50 %	20%	0.52	0.92	0.57

Table 6.3: Experimental Results from Head-On Scenario. *Compliant: successful avoidance as per procedure, semi-compliant: successful avoidance however not exactly per procedure, Failed: unsuccessful avoidance. Metric: path optimality key performance indicator (Eq. 2.7).*

Tests number two and three again suggest that the COLREG and critical regions surrounding the vessel require some fine tuning for the experimental environment. In both these cases, despite the collision being avoided, the system failed to detect and act upon conflict within the COLREG region, jumping immediately into emergency avoidance procedure. The total avoidance failures in tests four and seven were caused by localisation system failure. The issue being in both cases that the tracking algorithm incorrectly associated its position with the start of the path, causing a sudden jump backwards, leading to failed collision avoidance.

6.4.3. Overtake

The overtake scenario achieved the best results of all the scenarios with no failures in collision avoidance and 50% of tests yielding fully compliant avoidance (see Table 6.4). The scenario also yielded the best path optimality score of all, achieving an optimality metric of 0.33. Of the 50% of semi-compliant attempts, the main culprit preventing full compliance was the misidentification of the contact vessel as a static vessel. Although a static manoeuvre also yields an overtake procedure, semi-compliance was assigned due to the fact that the contact vessel should have been recognised as dynamic during the experiment as this would have influenced the outcome of the experiments, perhaps leading to conflict being detected sooner.



(a) Experimental Scenario and the Selected Local Path for Avoidance

```
-------------------------Guidance Update--------------------------
[INFO] [1604455418.142710]: COLREG conflict was detected
[INFO] [1604455418.143889]: AV has been assigned GiveWay status.
[INFO] [1604455418.144651]: Overtake avoidance mode activated.
[INFO] [1604455418.145916]: New local path has been assigned.
[INFO] [1604455418.146587]: Changing course to avoid collision.
```

(b) Procedural Response to the scenario from the Guidance System

Figure 6.21: Example Results from Overtake Experiment

Test Number	Collision Avoidance			Path Optimality		
	Compliant	Semi-compliant	Failed	Added Distance	Collision Vicinity	Metric
1		✓		0.20	0.81	0.26
2		✓		0.80	1.16	0.69
3	✓			0.40	1.8	0.22
4	✓			0.24	1.5	0.16
5		✓		0.69	0.8	0.86
6	✓			0.52	0.9	0.58
7	✓			0.50	2.1	0.23
8		✓		0.25	0.6	0.42
9		✓		0.38	1.50	0.25
10	✓			0.25	1.55	0.16
Average	50%	50 %	0%	0.42	1.27	0.33

Table 6.4: Experimental Results from Overtake Scenario. *Compliant: successful avoidance as per procedure, semi-compliant: successful avoidance however not exactly per procedure, Failed: unsuccessful avoidance. Metric: path optimality key performance indicator (Eq. 2.7).*

6.5. Conclusion

This Chapter has answered the final research question of this thesis: *How well do the guidance and navigation systems perform under experimental testing and what are their limitations?*. The performance of the perception and localisation branches of the Navigation System have been individually evaluated, providing an indication of the capabilities of stereovision sensors in the inland application. The collision avoidance performance of the Guidance system has been tested in both a simulated environment with controlled inputs and experimental an environment, where the data supplied by the Navigation System.

Perception testing provided promising results from both the robustness of obstacle detection and the accuracy of positioning. The obstacle detection procedure maintained high precision with increasing recall achieving a total precision of 97.4% and recall of 94.6%. The evaluation of the perception position accuracy yielded a root mean square position error of 0.31 metres. Localisation testing was conducted using data gathered under two path variants, a slalom/zigzag style path and a straight path. Within the workspace environment the position error metric was found to be 0.129m and the heading error to be 5.3°. A variation in localisation performance was however demonstrated along the path length, with the performance decreasing with an increase in length. The localisation system was however observed on multiple occasions to be susceptible to the impact of the kidnapped robot problem.

Testing of the collision avoidance procedure under simulation provided verification and validation of the Guidance System implementation, with fully compliant avoidance being achieved in all forty tests. However as the main aim of this thesis was to assess collision avoidance using stereovision technology, the most important results of this Chapter are from the experimental tests of the combined Guidance and Navigation System. Across all of the thirty tests conducted, an average avoidance success rate of 90% was achieved, meaning that 10% of interactions resulted in collision due to insufficient action. Of the successful tests however, only 40 out of the total 90% percent saw fully compliant avoidance with the remainder seeing either the premature activation of emergency procedure or the incorrect COLREG mode being activated.

The experimental testing has highlighted that whilst the Guidance System performed particularly well under simulation, with real world data supplied from the stereovision devices this same performance could not be accomplished. Despite almost always generating a path that successfully avoided collision with a contact vessel, experimental avoidance was not capable of consistently triggering the correct response due to the data from the navigation system not being reliable enough. The alteration of conflict region definitions could well provide a workaround solution to this issue as by expanding the COLREG region a greater opportunity for compliant regulation would be allowed. However that being said, the first port of call should be to address the data quality issues from the Navigation System.

7

Conclusion and Recommendations

This paper has presented the development of Navigation and Guidance Systems for an inland autonomous vessel. Reflecting back upon the main research question: *How can collision avoidance be achieved by autonomous inland vessels using stereovision?*, the main requirements of these systems were to enable collision avoidance on the inland waterway through the utilisation of stereovision sensor technology. The first section of this chapter will provide a summary of the work conducted and the second section shall thereafter cover the results achieved during evaluation. The final section of this chapter shall deliver proposals for the future research based upon the outcomes of this project and the observations made throughout.

7.1. Recapitulation

7.1.1. Standardisation

The first research question concerned the implementation approach with a the incentive to move towards an open source platform for autonomous inland vessels. It was concluded from the research that the Robot Operating System would provide the best middleware framework for the implementation of the Guidance and Navigation units. To enable future research developments within the ResearchLab Autonomous Shipping and the wider research community, it was also concluded from the literature review that Python offered the optimal programming language for the use case.

The implementation saw the use of ROS packages for the grouping of major tasks and the approach to software modularity continues even at a sub-package level. This approach was selected to promote future research development by allowing focus on the optimality of specific module tasks without the need to develop or revise an entire system. A total of four packages were developed, with one dedicated to guidance tasks and three being dedicated to the stereovision based navigation system. The separation of the navigation tasks allow for the modular application of these tasks independently of one another in other application cases.

7.1.2. Stereovision-based Navigation

The second research focus was on this stereovision-based Navigation System, with a multi-device system being proposed to achieve near range perception and localisation on board inland autonomous vessels. Four depth devices onboard enable the acquisition of both textural and depth data of the surrounding environment for perception tasks, covering the directions fore, aft, port and starboard of the autonomous vessel. A tracking camera, incorporating stereo fish-eye vision sensors and an integrated inertial measurement unit is utilised to provide pose data for autonomous vessel localisation.

The multi-device perception setup enables the vessel to maintain a constant lookout around the vessels in the collision risk regions. The utilisation a convolutional neural network to conduct object detection has highlighted its suitability to the autonomous inland vessel application. Not only does this approach provide advantages over arbitrary object detection but does so at impressive inference rates even when computationally limited to a small form-factor computer onboard a scale vessel. A refined approach to the acquisition of depth data avoids the unnecessary post-processing and manipulation of dense point clouds, which arguably add little value to a system centred around achieving collision avoidance.

The utilisation of a Visual Inertial Odometry approach provided a means of short range localisation using only stereovision and IMU data. The offloading of the task to a singular compact device, allows a significant boost to small scale applications and yet the VIO approach also lends itself well to larger application setups. Secondary processing of the perception and localisation data through the mapping procedure provides a coalescent output for the Navigation System, particularly well suited to provide sufficient information to ongoing collision avoidance tasks.

Under the given computational setup, the output of the Navigation System is capable of publishing outgoing data at rate of five hertz, which is considered sufficient for the low speed application. The perception and localisation sub-tasks alone were further found to be capable of reaching frequencies closer to ten hertz.

7.1.3. Inland Collision Avoidance

The third research focus resulted in a Guidance System being developed which incorporates a tailored collision avoidance protocol for the challenges of the inland waterway. A new configuration for conflict detection regions has been proposed to meet the challenges brought on by the architecture of the inland waterways and the nature of the interactions that take place on it. The techniques see the use of rectangular vessel domain regions whose dimensions are influenced by a function of the vessel speed. The autonomous vehicle is assigned two conflict regions, one that when infringed indicates a scenario whereby collision regulations can be followed and a second critical region where emergency avoidance procedure should be followed.

The collision resolution approach sees the integration of collision regulations and a secondary emergency procedure within a rule-based implementation compatible with the data available from a stereo-vision based navigation system. This two tier system prevents the AV from blindly following the collision regulations when a contact vessel is recognised to be non-compliant or an extremely close encounter requires additional action to be taken.

The local path planner incorporates a roll-out-trajectory generation technique that has been adapted and optimised for application on an inland autonomous vessel. The planner operates by generating candidate trajectories based upon the determined collision avoidance mode and the limitations of the navigable waterway channel. The optimal path is then selected from the candidate trajectories by applying three weight variants based upon the added distance, collision regulations and collision risk and solving the subsequent optimisation problem.

7.2. Evaluation

The perception solution was evaluated using experimental testing which focused upon the robustness of obstacle detection and the accuracy of positioning. Under the defined precision/recall key performance indicators, the obstacle detection procedure maintained high precision with increasing recall achieving a total precision of 97.4% and recall of 94.6%. However as the neural network was both trained and tested with one type of obstacle in one environment, the significance of this result is limited to the specific experimental environment.

The accuracy of obstacle positioning was evaluated using the ground truth from an OptiTrack camera system providing sub-millimetre accuracy. Data was gathered within the ten metre range covered by the perception system, with a total of 280 data points being gathered across three tests with varied vessel positioning. The comparison of the ground truth and perception system provided a quantification position error which yielded a root mean square position error of 0.31 metres.

Experimental testing to evaluate localisation performance was conducted using two path types, a slalom/zigzag style path and a straight path. The evaluation of the localisation procedure focused upon paths within a waterway length range of 15 metres as this was within the workspace bounds of the collision avoidance experiments. Along the full 15m length, the position error metric was found to be 0.129m and the heading error to be 5.3°. A variation in localisation performance was however demonstrated along the path length, with the performance decreasing with an increase in length. If one were to reduce the waterway length to 7.5m for evaluation sampling, the position and heading errors were seen to reduce to 0.068m and 2.6° respectively. These evaluations were made under normal operational cases, however the localisation system was observed on multiple occasions to be susceptible to the impact of the kidnapped robot problem.

Collision avoidance experiments provided an evaluation of the combined guidance and navigation system performance. Across all of the thirty tests conducted, an average avoidance success rate of 90% was achieved, meaning that 10% of interactions resulted in collision due to insufficient action. Of the successful tests however, only

40 out of the total 90% percent saw fully compliant avoidance with the remainder seeing either the premature activation of emergency procedure or the incorrect COLREG mode being activated.

The experimental testing has highlighted that whilst the Guidance System particularly well under simulation, with real world data supplied from the stereovision devices this same performance could not be accomplished. Despite almost always generating a path that successfully avoided collision with a contact vessel, experimental avoidance was not capable of consistently triggering the correct response due to the data from the navigation system not being reliable enough. The alteration of conflict region definitions could well provide a workaround solution to this issue as by expanding the COLREG region a greater opportunity for compliant regulation would be allowed.

The weakest link in the entire network during this combined testing appeared to be the localisation system which was prone to sudden and significant errors, whereby it would get lost in its environment, either associating itself with the wrong location or failing to recognise movement at all. Besides these sudden failures, the combination of the localisation position error under normal operation and the perception position error lead to the global state attributes, particularly contact vessel velocity to be assigned a misleading value.

7.3. Recommendation for Future Work

7.3.0.1. Guidance

A more sophisticated motion prediction procedure would allow for optimised interaction behaviour. The integration of COLREG based prediction would optimise reactions in head on scenarios. The current system selects a collision avoidance route under the assumption that the target vessel will continue on its current course, however in such a scenario the target vessel also has a responsibility to manoeuvre to its starboard side. Therefore, operating under the assumption that the target vessel is complying to COLREGs, the collision avoidance trajectory could be positioned closer to the global path, minimising the extra distance that needs to be travelled. Such assumptions would however require extra contingencies to be put in place to recognise and detect when a target vessel is failing to fulfil its roll under collision regulations.

As was mentioned during the presentation of the conflict regions, the assumption of vessel size based only upon the classification is somewhat limited. For vessel classifications whose dimensions can vary significantly in size, such as with freight barges, a less ambiguous approach would be desirable. A potential solution to this limitation would be to make use of the obstacle dimension which is supplied by the navigation system to estimate vessel size. However as this dimension is arbitrary it would first need to be associated with an actual vessel dimension. To do this, the orientation of the vessel needs to be known which could be obtained in several manners. One potential solution would be to train the neural network model to distinguish between the bow, stern and sides of a vessel. Or by assuming that vessel is orientated in the direction of its movement, the vessel's angle of orientation could be derived using the approach angle from the navigation system.

Whilst the navigation system is fully configured for handling multiple obstacles, the guidance system requires some adjustments in order to handle environments with multiple obstacles. The current setup has been configured assuming that only one obstacle is avoided at a time, which obviously does not hold for all scenarios. Furthermore testing was only conducted on a straight stretch of inland waterway. It would be beneficial to test the performance of collision avoidance when navigating along a curved waterway channel. It is anticipated that the simple motion prediction procedure and rigid conflict regions will perform unfavourably when navigating such obstacles. One potential solution may be to refactor the AV conflict region generator to follow the global path as opposed to generating the region inline with its current heading.

7.3.0.2. Navigation

As artificial intelligence will likely play a major role in the realisation of Autonomous Inland Vessels, a focus on data acquisition and model training would be a beneficial area of research. In the broader maritime environment, the Singapore dataset [53] and SeaShips [54] datasets provide image data and pre-trained object detection models for sea going vessels. Whereas to the best of the author's knowledge there exists no equivalent dataset specific to the inland waterway environment. Gathering the training data and configuring models for obstacles encountered on the inland waterway, from vessels to pontoons and buoys. Compiling such a dataset of annotated images could be an extremely valuable contribution to the developments of this research field.

An increase in onboard computational power could allow for a point cloud based SLAM algorithm, using data acquired by the depth sensors to be utilised. Fusing the results of this localisation solution with that of the existing

V-SLAM algorithm could improve localisation system performance without the system requiring additional short range sensor sets.

On the topic of sensor fusion, one of the major limitations of the stereovision-based navigation system is that it is not suitable for all weather conditions. In low visibility, the imaging sensors cease to provide the perception and localisation data required, meaning additional sensors would be required to support sailing in such conditions. Radar would be the primary choice to support navigation in low light and fog conditions, however as mentioned it is not without its accuracy and frequency limitations. This means that sailing under such conditions would lead to sub-optimal operational performance as contingencies would need to be expanded to account for reduced perception.

The challenges of autonomously mooring and navigating inland infrastructure such as locks will prove a particularly difficult task for a navigation system to handle. In further developments of the navigation system, it would be valuable to study the required sensor set to enable such levels of autonomy. It was discussed during the literature study that autonomous cars make use of ultrasonic sensors in combination with an array of imaging sensors in order to enable short range perception with sufficient accuracy. This aid autonomous cars in performing tasks such as parking and also supports the system during close range interactions with other vehicles. Perhaps a similar sensor set could be applied to aid autonomous inland vessels in handling close range interactions in narrow channels and assist in mooring procedures.

The current depth devices have a relatively limited horizontal field of view and a limited depth range. The newly released RealSense depth cameras would provide an improvement in both these specifications, with a field of view in the horizontal plane increasing by twenty degrees per device and the depth range increasing to twenty metres. This improvement in sensor technology could reduce the blind spots that exist in the current perception device configuration and the increased range make the application of the sensors to larger scale vessels feasible.

7.3.0.3. Control

Although the control system did not form a research focus of this paper, a few observations were made during experimental testing which present potential avenues for future research development. The control structure currently uses a path following technique whereby it simply navigates a path dictated by target waypoints that are continuously fed from the guidance system. Although a sufficient technique to provide the rudimentary change in trajectory needed for collision avoidance, in the absence of speed control the system did not have as much control over actuation as would be desired. The integration of a more advanced trajectory tracking technique could help the vessel in following the route planned by the guidance system. Furthermore, the use of a simple PID controller in handling the thruster angle did lead to a variation in path following performance along different path sections. Perhaps the integration of an adaptive PID strategy would be able to overcome such limitations.

Bibliography

[1] Bart van Riessen, Rudy R. Negenborn, and Rommert Dekker. Synchromodal container transportation: An overview of current topics and research opportunities. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9335:386–397, 2015.

[2] E. Verberght and E. Van Hassel. The automated and unmanned inland vessel. *Journal of Physics: Conference Series*, 1357(1), 2019.

[3] Port of Rotterdam. Port vision 2030. *Port of Rotterdam 2030*, (December), 2011.

[4] Christoph A. Thieme, Chuanqi Guo, Ingrid B. Utne, and Stein Haugen. Preliminary hazard analysis of a small harbor passenger ferry-results, challenges and further work. *Journal of Physics: Conference Series*, 1357(1), 2019.

[5] R. Groenveld, H. J. Verheij, and C. Stolker. *Capacities of inland waterways*. 2006.

[6] Michael Schuster, Michael Blaich, and Johannes Reuter. *Collision avoidance for vessels using a low-cost radar sensor*, volume 19. IFAC, 2014.

[7] Gerben Peeters, Marcus Kotzé, Muhammad Raheel Afzal, Tim Catoor, Senne Van Baelen, Patrick Geenen, Maarten Vanierschot, René Boonen, and Peter Slaets. An unmanned inland cargo vessel: Design, build, and experiments. *Ocean Engineering*, 201, 2020.

[8] Wei Wang, Banti Gheneti, Luis A. Mateos, Fabio Duarte, Carlo Ratti, and Daniela Rus. Roboat: An Autonomous Surface Vehicle for Urban Waterways. pages 6340–6347, 2020.

[9] Lifei Song, Houjing Chen, Wenhao Xiong, Zaopeng Dong, Puxiu Mao, Zuquan Xiang, and Kai Hu. Method of Emergency Collision Avoidance for Unmanned Surface Vehicle (USV) Based on Motion Ability Database. *Polish Maritime Research*, 26(2):55–67, 2019.

[10] Binghua Shi, Yixin Su, Chen Wang, Lili Wan, and Yi Luo. Study on intelligent collision avoidance and recovery path planning system for the waterjet-propelled unmanned surface vehicle. *Ocean Engineering*, 182(May):489–498, 2019.

[11] Lifei Song, Zhuo Chen, Zaopeng Dong, Zuquan Xiang, Yunsheng Mao, Yiran Su, and Kai Hu. Collision avoidance planning for unmanned surface vehicle based on eccentric expansion. *International Journal of Advanced Robotic Systems*, 16(3):1–9, 2019.

[12] Xiaojie Sun, Guofeng Wang, Yunsheng Fan, Dongdong Mu, and Bingbing Qiu. Fast Collision Avoidance Method Based on Velocity Resolution for Unmanned Surface Vehicle. *Proceedings of the 31st Chinese Control and Decision Conference, CCDC 2019*, pages 4822–4827, 2019.

[13] Yunsheng Fan, Xiaojie Sun, and Guofeng Wang. An autonomous dynamic collision avoidance control method for unmanned surface vehicle in unknown ocean environment. *International Journal of Advanced Robotic Systems*, 16(2):1–11, 2019.

[14] Jia yuan Zhuang, Lei Zhang, Shi qi Zhao, Jian Cao, Bo Wang, and Han bing Sun. Radar-based collision avoidance for unmanned surface vehicles. *China Ocean Engineering*, 30(6):867–883, 2016.

[15] Hongguang Lyu and Yong Yin. COLREGS-Constrained Real-Time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields. *Journal of Navigation*, 72(3):588–608, 2019.

[16] Bjorn Olav H. Eriksen, Erik F. Wilthil, Andreas L. Flåten, Edmund F. Brekke, and Morten Breivik. Radar-based maritime collision avoidance using dynamic window. *IEEE Aerospace Conference Proceedings*, 2018-March(March):1–9, 2018.

[17] Yan Wang, Wei Lun Chao, DIvyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q. Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:8437–8445, 2019.

[18] Tesla ©. Advanced Sensor Coverage. 2020. `https://www.tesla.com/autopilot`.

[19] Rui Song, Yuanchang Liu, and Richard Bucknall. Smoothed a* algorithm for practical unmanned surface vehicle path planning. *Applied Ocean Research*, 83:9 – 20, 2019.

[20] Yogang Singh, Sanjay Sharma, Robert Sutton, Daniel Hatton, and Asiya Khan. A constrained a* approach towards optimal path planning for an unmanned surface vehicle in a maritime environment containing dynamic obstacles and ocean currents. *Ocean Engineering*, 169:187 – 201, 2018.

[21] Linying Chen, R.R. Negenborn, and G. Lodewijks. Path planning for autonomous inland vessels using a*bg. volume 9855, pages 65–79, 09 2016.

[22] A. Bons, K. Molenmaker, M. van Wirdum, and C. F. Van der Mark. EconomyPlanner ; optimal use of inland waterways. *European Inland Waterway Navigation Conference*, 2014.

[23] Xavier Bellsolà Olba, Winnie Daamen, Tiedo Vellinga, and Serge P. Hoogendoorn. A method to estimate the capacity of an intersection of waterways in ports. *Transportmetrica A: Transport Science*, 15(2):1848–1866, 2019.

[24] Yamin Huang, Linying Chen, Pengfei Chen, Rudy R. Negenborn, and P. H.A.J.M. van Gelder. Ship collision avoidance methods: State-of-the-art. *Safety Science*, 121(April 2019):451–473, 2020.

[25] Michael R Benjamin. Autonomous COLREGS Modes and Velocity Functions. Technical report, MIT - Computer Science and Artificial Intelligence Laboratory, 2017.

[26] Ho Namgung, Jung Sik Jeong, Joo Sung Kim, and Kwang Il Kim. Inference Model of Collision Risk Index based on Artificial Neural Network using Ship Near-Collision Data. *Journal of Physics: Conference Series*, 1357(1), 2019.

[27] Sang Min Lee, Kyung Yub Kwon, and Joongseon Joh. A fuzzy logic for autonomous navigation of marine vehicles satisfying COLREG guidelines. *International Journal of Control, Automation and Systems*, 2(2):171–181, 2004.

[28] Guoge Tan, Jin Zou, Jiayuan Zhuang, Lei Wan, Hanbing Sun, and Zhiyuan Sun. Fast marching square method based intelligent navigation of the unmanned surface vehicle swarm in restricted waters. *Applied Ocean Research*, 95(December 2019):102018, 2020.

[29] Bjørn-Olav H Eriksen, Morten Breivik, Erik F Wilthil, Andreas L Flåten, and Edmund F Brekke. The Branching-Course MPC Algorithm for Maritime Collision Avoidance.

[30] Marcus Kotzé, Ali Bin Junaid, Muhammad Raheel Afzal, Gerben Peeters, and Peter Slaets. Use of Uncertainty Zones for Vessel Operation in Inland Waterways. *Journal of Physics: Conference Series*, 1357(1), 2019.

[31] Yogang Singh, Sanjay Sharma, Daniel Hatton, and Robert Sutton. Optimal path planning of unmanned surface vehicles. *Indian Journal of Geo-Marine Sciences*, 47(7):1325–1334, 2018.

[32] Shijie Li, Jialun Liu, Rudy R. Negenborn, and Feng Ma. Optimizing the joint collision avoidance operations of multiple ships from an overall perspective. *Ocean Engineering*, 191(May):106511, 2019.

[33] M. Łącki. Neuroevolutionary approach to colregs ship maneuvers. *TransNav*, 13(4):745–750, 2019.

[34] Ingunn Johanne Vallestad. Path Following and Collision Avoidance for Marine Vessels with Deep Reinforcement Learning. (June), 2019.

[35] Michael R. Benjamin, John J. Leonard, Joseph A. Curcio, and Paul Newman. A method for protocol-based collision avoidance between autonomous marine surface craft. *J. Field Robotics*, 23:333–346, 2006.

[36] Michael R. Benjamin, Henrik Schmidt, Paul Newman, and John J. Leonard. Nested autonomy for unmanned marine vehicles with moos-ivp. *J. Field Robotics*, 27:834–875, 2010.

[37] Hatem Darweesh, Eijiro Takeuchi, Kazuya Takeda, Yoshiki Ninomiya, Adi Sujiwo, Luis Yoichi Morales, Naoki Akai, Tetsuo Tomizawa, and Shinpei Kato. Open source integrated planner for autonomous navigation in highly dynamic environments. *Journal of Robotics and Mechatronics*, 29(4):668–684, 2017.

[38] International Maritime Organisation. Convention on the International Regulations for Preventing Collisions at Sea, 1972.

[39] Robert Pontius, Olufunmilayo Thontteh, and Hao Chen. Components of information for multiple resolution comparison between maps that share a real variable. *Environmental and Ecological Statistics*, 15:111–142, 06 2008.

[40] Zichao Zhang and Davide Scaramuzza. A Tutorial on Quantitative Trajectory Evaluation for Visual(-Inertial) Odometry. *IEEE International Conference on Intelligent Robots and Systems*, pages 7244–7251, 2018.

[41] R. Padilla, S. L. Netto, and E. A. B. da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237–242, 2020.

[42] Clint Nous, Roland Meertens, Christophe De Wagter, and Guido Croon. Performance evaluation in obstacle avoidance. pages 3614–3619, 10 2016.

[43] Tegawende F. Bissyande, Ferdian Thung, David Lo, Lingxiao Jiang, and Laurent Reveillere. Popularity, interoperability, and impact of programming languages in 100,000 open source projects. *Proceedings - International Computer Software and Applications Conference*, (August 2015):303–312, 2013.

[44] Martin Wende, Tim Giese, Serdar Bulut, and Reiner Anderl. Framework of an active learning python curriculum for first year mechanical engineering students. *IEEE Global Engineering Education Conference, EDUCON*, 2020-April:1193–1200, 2020.

[45] Abhinav Nagpal and Goldie Gabrani. Python for Data Analytics, Scientific and Technical Applications. *Proceedings - 2019 Amity International Conference on Artificial Intelligence, AICAI 2019*, pages 140–145, 2019.

[46] Intel®. Intel® RealSense™ Camera D400 series Product Family Datasheet, 2019.

[47] Anders Grunnet-Jepsen and Dave Tong. Depth Post-Processing for Intel ® RealSense ™ D400 Depth Cameras.

[48] M. Hepworth. Obstacle Perception, Localisation & Tracking for Autonomous Inland Vessels using Stereoscopic Depth Vision. Technical report, TU Delft, Delft, 2020.

[49] Anders Grunnet-Jepsen, Michael Harville, Brian Fulkerson, Daniel Piro, Shirit Brook, and Jim Radford. Introduction to Intel® RealSense™ Visual SLAM and the T265 Tracking Camera, 2020.

[50] Intel®. Intel RealSense ™ T265 Tracking Camera, 2019.

[51] Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. 2011.

[52] Pieter Kingma, Mattihjs Roebroek, Daniel Blommestein, Zasha van Hijfte, and Pim Hofste. Trajectory tracking for Grey Seabax. 2020.

[53] D. K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek. Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 18(8):1993–2016, 2017.

[54] Z. Shao, W. Wu, Z. Wang, W. Du, and C. Li. Seaships: A large-scale precisely annotated dataset for ship detection. *IEEE Transactions on Multimedia*, 20(10):2593–2604, 2018.

A

Appendix A - Scientific Research Paper

Collision Avoidance for Autonomous Inland Vessels using Stereovision*

Matt Hepworth*[1], Vasso Reppa[1], Vittorio Garofano[1], Rudy Negenborn[1]

***Abstract*—This paper explores an approach to collision avoidance for autonomous vessels on the inland waterway. The potential role of stereovision technology as a primary sensor set is investigated with the development of a multi-device Navigation sub-system setup for mid-range perception and localisation tasks. The specific performance of these sensors in the application is evaluated with experimental testing, demonstrating their capabilities and limitations. Attention is further applied to the guidance sub-system of an autonomous vessel. Techniques for conflict detection and conflict resolution suitable for the inland waterway environment are proposed, with the integration of collision regulations and emergency contingency protocols. Collision avoidance procedure is independently evaluated with simulation and thereafter experimental collision avoidance performance is tested with the combined inland Guidance and Navigation subsystems.**

***Index Terms*—Autonomous Shipping, Autonomous Surface Vehicle, Stereoscopic Vision, Artificial Intelligence, Inland Waterways, Collision Avoidance**

I. Introduction

The inland waterway once enabled an industrial revolution, yet the emergence of coalescent road networks has seen its true worth be all but disregarded. Despite the amenity of unimodal travel being compelling, growing awareness for sustainability has reignited interest in more fuel efficient modalities for transportation. Traffic congestion remains a persistent issue in urban areas and logistical hubs and fixed infrastructure links do not always offer expedient options to cross waterways. Increased utilisation of the inland waterways could offer a solution to achieving emissions targets, easing congestion and providing alternative crossing solutions. Attaining said utilisation and future-proofing the market share of the inland waterway does however require innovative solutions.

Concepts such as synchromodality have presented approaches to enable a modal shift in hinterland freight transportation through the creation of an interconnected, integrated and cooperative freight transportation network [1]. However its adaptive nature demands high efficiency and reliability from the inland waterway network. Autonomous shipping has the potential to increase efficiency and reliability and will arguably play a major role in the evolving transport revolution, returning the transport modality to its former glory.

Achieving autonomy is no mean feat and arguably the most critical challenge to an autonomous inland vessel (AV) is the suitable execution of collision avoidance. In the absence of the usual human operator, an AV requires a Guidance, Navigation and Control System (GNC) to assume responsibility for executing this task. Within this paper a main focus shall be applied to the Guidance and Navigation branches. The Guidance subsystem is responsible for, amongst other tasks of generating a local path to avoid an obstacle. The Navigation sub-system uses sensor and/or proprietary communication data such as AIS to establish the current position and state of the AV as well as evaluating the obstacles in its environment and their attributes.

Citation	Sensors †	Avoidance ‡	COLREGs	Evaluation
[2]	G, I, L, C, A	IvP	No	Experiment
[3]	I, L, C	RTG	No	Experiment
[4]	R	VO, Re	Yes	Simulation
[5]	R, A	Re	Yes	Simulation
[6]	A, R	VO	Yes	Simulation
[7]	A, R	MPC	Yes	Both
[8]	G, R, A, M	CC	No	Simulation
[9]	R	VO	No	Experimental
[10]	A, R, G	APF	Yes	Simulation
[11]	R	DW	No	Experimental

TABLE I: Maritime Collision Avoidance.
† *G: GPS, I: IMU, L: LiDAR, C: Camera, R: Radar, A: AIS*
‡ *IvP: Interval Programming, RTG: Rollout Trajectory Generator, VO: Velocity Obstacle, Re: Re-planning, MPC: Model Predictive Control, CC: Collision Cone, APF: Artificial Potential Field, DW: Dynamic Window*

In Table I, it can be seen that a host of sensors are typically utilised in order to enable collision avoidance. Radar, GPS and AIS are the primary choices due to their proven, dependable reputation from decades of being used as navigation aids. Whilst offering suitable coverage for collision avoidance in open-water environments, inland vessels tend to be subject to closer interaction between vessels, meaning that they require additional sensors to achieve near range perception and localisation. The inland applications in the first two rows of the table [2][3] both utilise LiDAR and stereovision to provide this supplement, however in both cases stereovision takes a back seat role.

Autonomous Inland Vessels perhaps exhibit perception requirements closer to those of autonomous ground vehicles as opposed to their sea-based counterparts. In the automotive industry, LiDAR has also been the most popular sensor choice for achieving mid range perception and was long considered a necessity. However deviation from this preference is emerging with manufacturers turning instead to imaging sensors due to the affordability and expanding capabilities of computer vision. A study by researchers at Cornell University has further gone on to prove the capability of stereovision to conduct the tasks previously assumed only possible using LiDAR [12]. Whilst inland vessels are arguably not restricted by the

* This work was supported by the Researchlab Autonomous Shipping (RAS) at Delft University of Technology
[1] Department of Maritime and Transport Technology, Delft University of Technology, The Netherlands, 2628 CD Delft, The Netherlands
* Corresponding author

same necessity for consumer accessibility, reducing the initial investment cost has the potential to catalyse adoption as well as benefit from the benefits of computer vision advancements.

The typical procedure for collision avoidance is well covered in a comprehensive review of the state-of-the art by Huang et al. [13]. A useful breakdown is provided of the typical sub-tasks involved, which are neatly divided into motion prediction, conflict detection and conflict resolution.

The incorporation of radial safety regions or vessel domains during conflict detection tasks are popular in maritime solutions to autonomy [4][5][6][7][8][14][15]. Vessels travelling on the narrow channels of inland waterway networks are however frequently subject to close side-side interactions with one another. This renders the direct application of radial regions less suitable, particularly in the case of larger vessels. The elliptical regions presented in [16] or the rectangular uncertainty zones presented in [17] may perhaps inspire more eloquent solutions to autonomous inland vessel applications.

Conflict resolution represents the cornerstone of collision avoidance. Its responsibility lies in determining a suitable collision free path based upon all the information available from preceding tasks. Artificial potential field, collision cone and velocity obstacle approaches are some of the most frequently encountered techniques for managing local routing [18]. Yet of late, research into maritime collision avoidance has been further advancing in its maturity. More recent work includes the proposal of hybrid solutions to local planning [4] [6] [16] and emergency contingency when normal avoidance is not possible [5]. Moreover, focus has even turned away from the generation of merely feasible avoidance and towards optimised avoidance by using a rolling horizon technique to select optimal heading angles [19]. Advancements in artificial intelligence have also lead to their consideration in solving the problem of maritime collision avoidance [20][21][22].

Many of the solutions for maritime collision avoidance have been adopted from other sectors. In some cases, these solutions have seen adaption and evolution prior to implementation, in other cases they have been more directly implemented. For example, the open source local path planning technique of [23] designed for ground vehicles has been utilised for inland surface vehicles on the canals of Amsterdam [3].

The importance of COLREG integration within the conflict resolution stage is clearly apparent. To this end multiple techniques have been proposed to integrate COLREG compliance into the local planner. Fuzzy rules have been integrated into a modified Virtual Force Field approach [14] and rule based methodologies that in some cases consider the fact that compliance is not a binary problem [24].

In this paper, an approach to achieve collision avoidance for autonomous inland vessels using is implemented. In section II a stereovision-based Navigation System is introduced. Following which, a strategy for collision avoidance is introduced for the inland environment (Section III). An evaluation of the Guidance and Navigation Systems is conducted in Section IV with a particular focus on the experimental performance of the stereovision sensors as well as the execution of collision avoidance. Finally, a conclusion is delivered in Section V, with recommendations for future research directions.

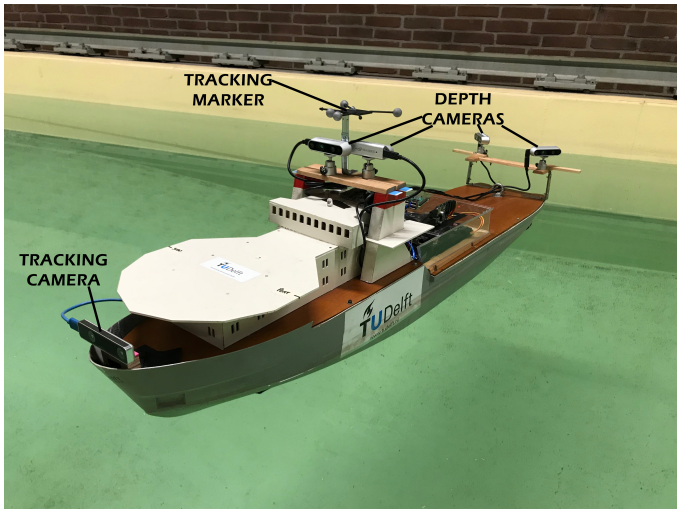


Fig. 1: The Grey Seabax Test Vessel.
Tracking Camera: stereovision device for localisation tasks. Depth Cameras: stereovision devices used for perception tasks. Tracking Marker: passive OptiTrack© marker to track vessel position during evaluation (ground-truth measurement).

II. Stereovision Based Navigation System

A fully operational navigation system of a vessel must have the capacity to perform perception and localisation under any condition of visibility to comply with collision regulations. This paper will narrow its focus and review the performance of stereovision in clear, well lit conditions. It is recognised that during low light conditions and/or poor weather conditions, the developed system would require at least one secondary sensor set to supply sufficient data. Furthermore, the navigation system of an autonomous vessel will likely make use of sensor fusion to improve accuracy and reliability in all sailing conditions and varying circumstances. A thorough understanding of the advantages and limitations of each individual sensor set is consequently imperative for optimal equipment selection and the fusing of sensor set data.

A. Perception

Rule five of the collision regulations states that a vessel must maintain a proper look-out at all times so to make a full appraisal of the situation and the risk of collision. Whilst a human operator will never be able to maintain a look-out in all directions surrounding the vessel, at all times, their vision does not have a fixed orientation, providing flexibility in field of view. When perceiving the environment using stereovision devices, such flexibility is not achievable in the absence of a rotating mount, such as those used by LiDAR and Radar sensors. Nonetheless, the affordability of stereovision technology enables the use of a multi-device setup, which could arguably achieve a more consistent lookout than any human operator ever could, whilst also not being susceptible to the mechanical failures as a rotating sensor.

Whether a perception system can be considered to fulfil the expectations of a proper look-out is open to interpretation, as are scenarios that present collision risk. Due to the architecture

of an inland waterway, vessels generally travel along a central path of the channel section in a manner synonymous to travel on road networks. Focusing perception ahead of the vessel, in its direction of movement where collision risk is higher and where opportunity for avoidance is present can thus be considered a more suitable utilisation of the sensor.

A proposed camera configuration for the test vessel can be visualised in Figure 2 and the mounted depth cameras can be seen in Fig. 1. A total of four Intel Realsense D345i depth devices are utilised covering a forward weighted coverage around the vessel. These devices have a depth range of ten metres, which is sufficient for the scale vessel in question, offering a coverage of magnitude over seven times the vessel length. For larger scale applications, device selection or custom stereovision setups would need to consider the specific requirements of the vessel scale, with the imaging sensors and the baseline influencing the range of a stereovision sensor.

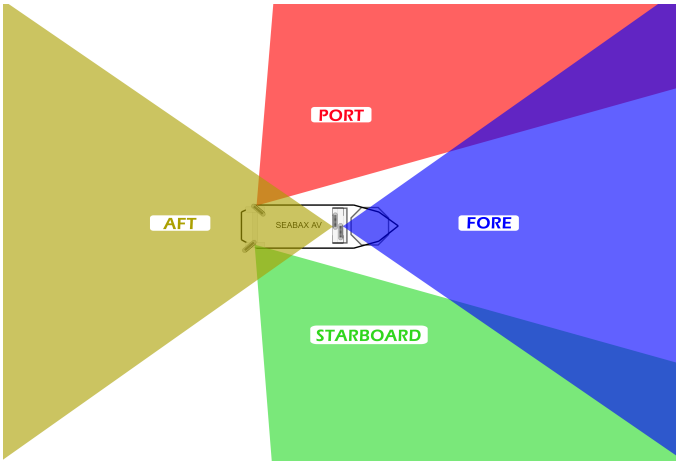


Fig. 2: Perception Sensors - Field of View

The depth cameras provide both point cloud data through a depth frame extracted from two infrared sensors and a colour frame from a separate RGB sensor. The colour frame is used for obstacle detection, with the resultant region of interest (ROI) being used to extract relevant position data from the depth frame. As the colour and depth frames are not physically aligned in the device, alignment must be conducted during post-processing. Sub-sampling is applied to the depth frame directly prior to this alignment by decimating the depth frame through a factor of two both reduces subsequent computation by a factor of four and has the added benefit of smoothing the depth data through the removal of dead depth pixels. Additional processing such as spatial and temporal filters were deemed unnecessary for the application requirements.

The potential of an artificial intelligence approach to maritime obstacle detection is well founded [25] [26] [27] and this potential shall be further explored in this implementation. A Convolutional Neural Network, with the MobileNet Single Shot Detector v2 framework is utilised for the specially trained Tensorflow model. The neural network model should be trained to recognise all obstacles that may be pose a collision risk in the interaction waterway environment. At this stage of research, the model is only trained to recognise one object, specifically the model tugboat that will be used in the experimental collision avoidance testing.

Inference is run on the Jetson TX2 mounted onboard the vessel, with frames being processed from all four camera streams. In order to maintain performance under this heavy processing burden, the trained model is converted to a Universal Framework Format (UFF) and subsequently parsed to build a 16-bit optimised TensorRT engine. Running inference on the TensorRT engine model was found to reduce latency fivefold over the original Tensorflow format with the max execution rate per frame being reduced from 125ms to 25ms. This enables each of the four perception device streams to be processed at up to 10Hz and does so without exhibiting a reduction in average precision. The performance of the trained model shall be further evaluated in Section IV and an example detection output stream can be visualised in Fig. 3, with the key outputs being the obstacle bounds and classification (C_{type}).

Fig. 3: Obstacle Detection Output

The obstacles detected in the colour frame must subsequently be analysed to determine their actual position in the 3D space. Each obstacle bound defines a rectangular area within the frame where an obstacle has been detected and due to frame alignment, this Region-of-Interest (ROI) also corresponds to the same frame area within the depth frame. Using the device intrinsics to assess the ROI within the depth frame, allows for point cloud points to be defined. The Closest Obstacle Point (COP) is then extracted from the point cloud data within the ROI and is taken forward as the reference point of obstacle location. The position at this stage is defined in coordinates relative to the device within which the obstacle has been detected. The device coordinates of obstacles are translated into one homogeneous reference frame during the mapping stage.

The perception system is built within a ROS package that is run on the onboard Jetson TX2 and its output is published over the ROS network at a frequency of 5Hz. The message that is published provides a list of obstacles and their corresponding attributes. The attributes that are published for each object are device position, obstacle classification, obstacle device coordinates and the obstacle's lateral dimension. This implementation approach allows for the perception package to be directly reapplied to other vessels, even ones that use a different number and orientation of depth devices.

B. Localisation

To localise the Autonomous Vessel, a dedicated tracking device is utilised which makes use of two fish eye lenses and a built in inertial measurement unit to conduct stereo V-SLAM (Visual- Simultaneous Localisation and Mapping). The device selected for this application also belongs to the Intel RealSense range, with the T265 camera fulfilling this role.

The tracking camera is mounted to the bow of the vessel as seen in Fig. 1 using a rigid 3D printed bracket. The position at the bow of the vessel was selected to minimise occlusions to the vision sensors and IMU noise due to vibrations at this location were not encountered. Although SLAM algorithms could also be run using data from the perception devices, offloading this task to an external device drastically reduces computational load on the on-board processor. The SLAM algorithm that runs onboard the T265 uses a Visual-Intertial-Odometry (VIO) technique, which relies upon data from two fish eye lenses and an IMU. These fish-eye sensors allow a large field of view to be covered, whereby more features can be detected and tracked within in the environment and provides a calibrated scale of the environment even with a single sensor set. The fusion of visual feature recognition with high frequency IMU data allows for the VIO algorithm to track movements over a shorter time steps, enhancing the tracking accuracy.

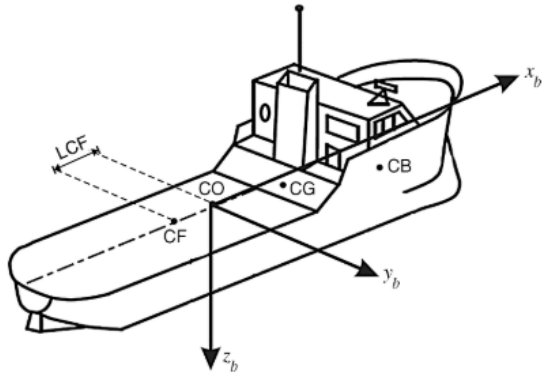


Fig. 4: Body Reference Frame [28]

The only post-processing tasks that are conducted by the localisation system concern the translation of the device coordinate axes to correspond with the body reference frame and the extraction of Euler angles from the quaternion orientation data. The position of the vessel is only required within two body-frame dimensions, x_b and y_b as defined in Fig. 4 and the Euler angle of main interest is the yaw angle (ψ) which provides a means to derive vessel heading. The output after post-processing was found to consistently achieve a frequency of 10Hz onboard the Jetson TX2 and is published over the ROS network, with the localisation system to being built within a ROS package to allow for direct application to other vessels in future research.

C. Mapping

Mapping represents the final sub-task in the Navigation system and is responsible for the post-processing of the results from the perception and localisation sub-tasks to provide useful data for collision avoidance tasks. The processing concerns two main procedures, firstly the translation of perception results to map positions into a single homogeneous reference frame, and secondly the fusion of localisation and perception data to map the position of the AV and obstacles in the global coordinate system.

Each obstacle position attained by the perception system is translated from the individual depth device coordinate system into a consolidated AV domain. The origin of the AV domain, is located at the geometric centre of the vessel as indicated by point CO in Fig.4, with the body axes orientation also matching that of the 2D AV coordinate system. Translation involves applying the displacement of the relative camera from this origin point and accounting for the configured camera angles through axes rotation. The result is a set of obstacle coordinates within the AV domain (X_{AV_D}, Y_{AV_D}). One particularly useful attribute for collision avoidance tasks is the obstacle bearing and can be defined as seen in Eq. 2.

$$\hat{\beta} = atan2(Y_{AV_D}, X_{AV_D}) \cdot \frac{180}{\pi} \tag{1}$$

$$\beta = \begin{cases} \hat{\beta}, & \hat{\beta} \geqslant 0^\circ \\ 360 + \hat{\beta}, & \hat{\beta} < 0^\circ \end{cases} \tag{2}$$

With the obstacles having been coalesced into a singular coordinate frame, any replicated obstacles that exist in the overlapping field-of-view regions are eradicated as per Algorithm 1. After which, each obstacle can be associated with an identification number for tracking over time. Tracking is achieved using a Global Nearest Neighbour technique whereby the coordinates of each obstacle detected in the current iteration are compared with all obstacle positions in the previous iteration and associated by minimising Euclidean distance. To enable user-interaction with the navigation system when necessary, a visual representation of the obstacles mapped and associated within the AV domain is provided in the form of a pseudo-radar plot as can be seen in Fig. 5.

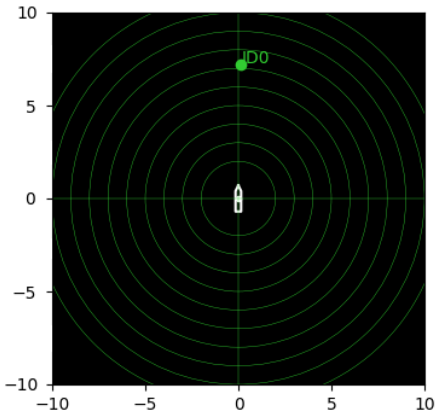


Fig. 5: Pseudo-radar Obstacle Monitor

Global mapping is conducted by utilising the localisation data. The global coordinate system requires a base map to be provided by the user that details the area of navigation, the starting position of the AV and the start heading (φ_0). The AV position (X_{av}, Y_{av}) can with this be defined by summing the start position with the position data from the localisation output. By monitoring the position over time, the AV speed (u_{av}) can be determined, following a constant time step of one second. The relative heading value is calculated through summing the initial offset with the yaw value (ψ) as defined in Eq. 3 and then applying the cases in Eq. 4.

$$\hat{\varphi} = \varphi_0 + \frac{180\psi}{\pi} \tag{3}$$

$$\varphi = \begin{cases} \hat{\varphi}\ , & 0^\circ \leqslant \hat{\varphi} \leqslant 360^\circ \\ 360 + \hat{\varphi}\ , & \hat{\varphi} < 0^\circ \\ \hat{\varphi} - 360, & \hat{\varphi} > 360^\circ \end{cases} \tag{4}$$

Defining the global coordinates of each obstacle (X_{cv}, Y_{cv}) is accomplished through the translation of obstacle coordinates in the AV domain using the current AV position and orientation. Using the obstacle IDs from the previous section, obstacles can also be tracked over time to define their state. The velocity vector generated between the obstacle position at the previous time step (one second) and the current position can be referred to as the obstacle's relative track. The track vector can be described by its magnitude and direction. The direction defined in Eq. 6 being referred to as the Track Angle (τ) and the magnitude describing the vessel speed (u_{cv}). To assist in collision avoidance procedure it is also useful to know the angle at which a dynamic obstacle is moving relative to the AV itself. This can simply be determined by finding the difference between the heading of the AV and the track angle of the obstacle. This directional measure of obstacle dynamics shall be referred to as the approach angle (α) and is defined in Eq. 8.

$$\hat{\tau} = (atan2(\dot{Y}_{cv}, \dot{X}_{cv}) + \varphi_0) \cdot \frac{180}{\pi} \tag{5}$$

$$\tau = \begin{cases} \hat{\tau}\ , & 0 \leqslant \hat{\tau} \leqslant 360^\circ \\ 360 + \hat{\tau}\ , & \hat{\tau} < 0^\circ \\ \hat{\tau} - 360, & \hat{\tau} > 360^\circ \end{cases} \tag{6}$$

$$\hat{\alpha} = \tau - \varphi \tag{7}$$

$$\alpha = \begin{cases} \hat{\alpha}, & \hat{\alpha} \geqslant 0^\circ \\ \hat{\alpha} + 360, & \hat{\alpha} < 0^\circ \end{cases} \tag{8}$$

The mapping sub-task is also embedded within a ROS package and is responsible for providing the output of the Navigation System. The mapping procedure runs on the host PC (Macbook Pro A1278) and the output publication rate can maintain a frequency of five hertz, which is considered sufficient for the low speed application. The specific output in this case is especially suited for the collision avoidance procedure that shall be detailed in the following chapter. The output is again published over the ROS network and contains two topics, a summary of which is given below in Table. II. One of these topics streams attributes of the AV and the other details the attributes of the obstacles, which from this point on shall be referred to as contact vessels.

Attribute	**AV Data**	**CV Data**
Global Coordinates	(X_{av}, Y_{av})	(X_{cv}, Y_{cv})
AV Domain Coordinates	-	(X_{AV_D}, Y_{AV_D})
Speed	u_{av}	u_{cv}
Heading	φ	-
Bearing	-	β
Approach Angle	-	α
Track Angle	-	τ
Classification	-	C_{type}
Identification Number	-	ID

TABLE II: Output of Navigation System

III. Collision Avoidance for an Autonomous Inland Vessel

The Guidance System of an Autonomous Inland Vessel requires a local path planner capable of handling collision avoidance. The procedure for collision avoidance can be aptly divided into the topics of motion prediction, conflict detection and conflict resolution s per the review of Huang et al. [13]. The term Contact Vessel (CV) describes an obstacle classified as a vessel with which the AV is in a collision interaction. The procedure within this section only covers interactions between the AV and a single contact vessel.

A. Motion Prediction

At this initial stage of research development, a physics based model is utilised with the implementation of more sophisticated techniques being left to form the focus of future research. Through the assumption that a contact vessel is holonomic and can move freely in the horizontal plane, the vessel's motion can be predicted over coming time steps using its current speed and track. Extending the current velocity vector to span a range future time intervals provides a prediction of future motion, albeit a crude one. Both the speed (u_{cv}) and the track angle (τ) of the contact vessel are supplied by the navigation system with their relations to the movement in the horizontal plane being defined in Equations 9 and 10 with the time step (t) being defined in seconds.

$$\dot{x}(t) = x(t) - x(t-1) = u \cdot cos\psi \tag{9}$$

$$\dot{y}(t) = y(t) - y(t-1) = u \cdot sin\psi \tag{10}$$

The most useful value that can be supplied by motion prediction is an estimate of future position so to define the fore bound of a vessel's conflict region and assess collision risk. The predicted position (x_{pr}, y_{pr}) of the contact vessel can be found by applying the holonomic model as defined in Eq. 11 and 12, with t_0 as the current time step. The motion prediction span t_{pr} must be suitably configured for the application to ensure that sufficient warning is provided

whilst not creating an overly sensitive avoidance system. For the model tugboat that will assume the role of the contact vessel during experimental testing, a prediction span value of three seconds is assigned.

$$x_{pr} = x(t_0) + (t_{pr} \cdot u_{cv} \cdot cos\tau) \tag{11}$$

$$y_{pr} = y(t_0) + (t_{pr} \cdot u_{cv} \cdot sin\tau) \tag{12}$$

B. Conflict Detection

Conflict regions in maritime autonomy are typically defined in radial form, which whilst valid for open-water applications are less applicable to the narrow channels of inland waterways. As such, rectangular conflict regions will be proposed in this paper, inspired to a degree by the uncertainty zones presented in the Hull-2-Hull research project [17].

In this approach, each contact vessel is assigned a singular conflict region. The conflict region of a static CV is defined to be one times the beam of the vessel around all edges. When a CV is moving, only the forward conflict region bound changes, with an extension (X_{con}) being added to the static region in this direction. This extension is defined in Eq. 13 using the motion prediction model. The static and dynamic regions of the CV can be visualised in Fig. 6. Although largely applicable, the definition of the static region based upon vessel beam may not be well suited to large vessels operating in particularly narrow waterway stretches. The beam and length of the contact vessel are retrieved from a database which lists all vessel classifications that the neural network model can recognise and the approximate dimensions of said vessels.

$$X_{con} = t_{pr} \cdot u_{cv} \tag{13}$$

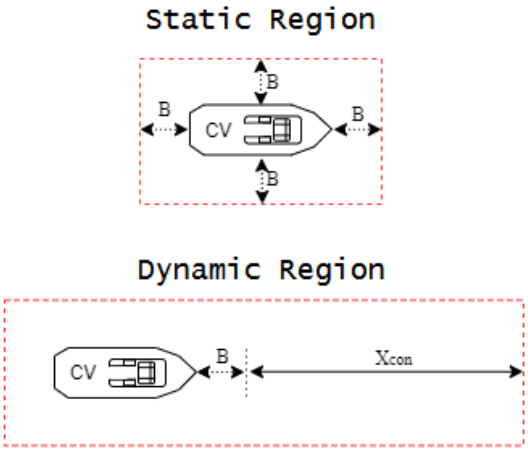


Fig. 6: Contact Vessel Conflict Region. *Where B is the beam of the vessel and X_{con} is the forward bound extension.*

A different procedure is followed for the assignment of the AV conflict regions as to that used for the contact vessel. The AV requires two regions, one larger conflict region to trigger generic collision avoidance which will be referred to as the COLREG zone and one smaller region intended to incite emergency collision avoidance procedure which will be referred to as the critical zone. Fig. 7 illustrates how the configurations of the AC conflict regions.

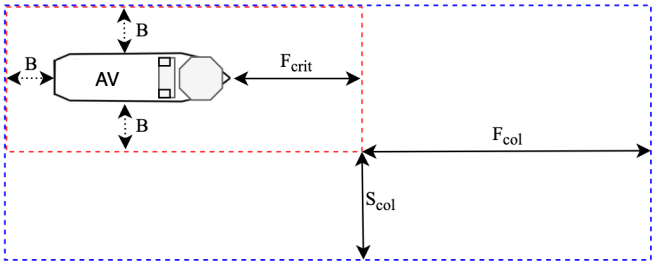


Fig. 7: Autonomous Vessel Conflict Regions. *Blue-dashed: COLREG region, red-dashed: critical region. F_{col} describes the fore COLREG bound, S_{col} the starboard COLREG bound and F_{crit} the fore critical bound. B refers to the beam of the vessel.*

The critical conflict region surrounds the port, starboard and aft edges at identical distances, defined at this stage as being one times the beam of the autonomous vessel (B_{av}). This selection also accounts for the port and starboard bow blind-spots of the AV's perception system. The size of the critical region ahead of the autonomous vessel is defined by considering its deceleration rate, to account for a worst case scenario. The theoretical stopping distance (d) can be defined by the heuristic relation between velocity (u_{av}) and deceleration rate (a_{av}) in Eq. 14. The value F_{crit} seen in Fig. 7 is subsequently driven by this stopping distance as defined in Eq. 15, with a safety factor of 1.25 being applied. Whilst applicable to the test vessel used in this study, the use of the stopping distance could prove less well suited to vessels with a high inertia.

$$d_{stop} = \frac{u_{av}^2}{2a_{av}} \tag{14}$$

$$F_{crit} = B_{av} + 1.25 \cdot d_{stop} \tag{15}$$

The COLREG zone surrounding the AV is not uniform as can be seen in Fig. 7, rather the region extremities are weighted in the fore and starboard directions to support adherence to regulation scenarios where the AV can be designated give-way status. The COLREG zone remains identical to the critical zone along the port and aft edges, due to vessel interactions on these sides yielding the AV to assume the role of the stand-on vessel. The region dimensions in the fore and starboard directions edges are driven by the vessel speed and a defined time period as defined in Eq. 16 and 17 which provide a distance in metres. For the Seabax AV, t_f is set to six seconds and t_s to three seconds with minimum dimension results being limited to 1.2 and 0.6 metres respectively.

$$F_{col} = u_{av} \cdot t_f \tag{16}$$

$$S_{col} = u_{av} \cdot t_s \tag{17}$$

The detection of conflict involves the constant monitoring of the two AV conflict regions, the COLREG zone and the critical zone. At each iteration it is assessed whether the conflict region of a contact vessel is infringing upon one of the AV regions. If a conflict is detected with a contact vessel, a trigger is set for the conflict resolution procedure to assess the situation and act accordingly. This trigger indicates not only that a conflict has been detected but also the protocol, be it COLREG or critical depending on which AV region has been infringed.

C. Conflict Resolution

A predominantly rule-based collision avoidance procedure is followed by the local planner at this stage of development. The avoidance strategy presented focuses on single vessel interactions within an environment typical of a simple inland waterway channel. The resolution procedure is triggered when a conflict is detected and the protocol has been set, with the primary protocol requiring a collision regulation procedure to be followed and the secondary protocol inciting emergency collision avoidance. Two stages exist within conflict resolution with avoidance procedure being addressed first and the local path generation thereafter.

The avoidance procedure is responsible for determining the way in which the AV resolves conflict. Its purpose is to determine the status and mode of avoidance. The avoidance status details the responsibility of the AV in a given interaction and the avoidance mode identifies the type of interaction that is taking place, which indicates the specific action that should be taken by the AV to avoid a collision. The type of conflict that has been detected has an influence upon the avoidance procedure that is followed. When a COLREG conflict is detected, avoidance is conducted based upon maritime collision law and when a critical conflict is detected, an emergency procedure is followed in a final attempt to avoid collision. The avoidance procedure uses the speed, bearing and approach angle data from the Navigation System to inform its decisions.

Algorithm 2 defines the the logic procedure for determining COLREG state where the main three COLREG interactions, head-on, overtake and crossover are considered. In any of these scenarios the AV can either be allocated a stand-on or give-way status, with a specific avoidance mode being allocated for all give-way cases. For all cases where a stand-on status is assigned to the autonomous vessel, it shall remain on its course as per regulations. Whilst maintaining this course, the critical region should continue be monitored and in the case where the contact vessel infringes this region, the COLREG compliance can be dropped in favour of emergency avoidance. If a critical conflict is detected, the emergency procedure in Algorithm 3 is followed to protect the AV from collision as far as possible. If a collision is anticipated yet it is still feasible for avoidance action to be taken, a hard turn to the port or starboard turn can be triggered to avoid the contact vessel..

Acting upon the determined avoidance mode requires the generation of a local path that suitably circumnavigates the obstacle. Not all of the existing maritime solutions to collision avoidance are suitable for application on the inland waterway and in many ways architecture of the inland waterway bears a closer resemblance to road networks. This paper presents an an implementation of a Roll-out Trajectory Generation technique for the inland waterway, adapted from an approach developed for autonomous cars [23].

The generation of candidate trajectories works simply by comparing the current position of the AV and the navigable edge of the waterway channel to define a bound and then creating trajectories that run parallel to the current path, with spacing between one times the width of the AV. As the avoidance mode indicates the direction in which the AV needs to travel, the candidate trajectories need only be generated to one side, most often astarboard. The number of trajectories that can be generated to the side of the AV is defined in Eq. 18, where d is the distance to the edge of the waterway from the current path and B_{av} is the AV beam.

$$\lfloor n \rfloor = \frac{d - B_{av}}{B_{av}} \tag{18}$$

A visual example of candidate trajectory generation can be seen in Fig. 8. Each trajectory is generated at a parallel distance of one times the vessel beam from the global trajectory or the superseding candidate trajectory. Smoothing the roll-out trajectory itself is generally neglected as a clear indication of manoeuvre intention is required by regulations and waypoint progress incorporates an acceptance radius which in itself offers an element of path smoothing. That being said, waypoint generation does require an element of path smoothing when the generated trajectory is close to the the edge of the waterway. Candidate trajectory three in Fig. 8 can be seen to have a smoothed roll-out region so to ensure the vessel has adequate opportunity to change its course to meet the parallel path.

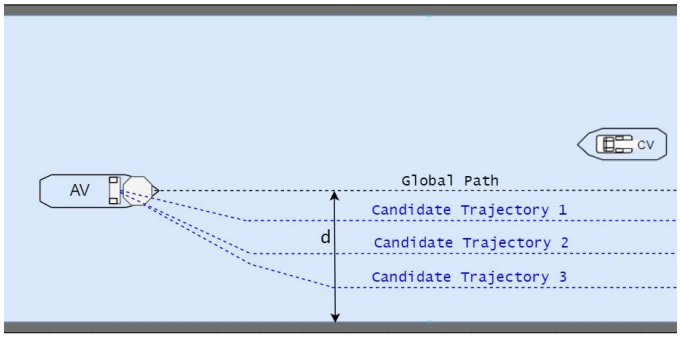


Fig. 8: Rollout Trajectory Generation

The path selection procedure determines the most suitable route from the candidate trajectories and is only required for the COLREG protocol as in an emergency, only one trajectory is generated. To select the best candidate trajectory in a given interaction, integer weights are assigned to each path and a simple optimisation problem is solved to minimise cost. Three influential factors were identified for the inland application and these regard the distance efficiency of a path, the expected avoidance action based upon CORLEGs and the risk of collision that a path carries.

The distance weight (d_i) is added to each of the trajectories based on the distance from the current path in an order

ascending magnitude with distance as shown in Table. III. The COLREG weight (c_i) can also be found in this table and is assigned based upon the avoidance mode that has been set which indicates the magnitude of expected avoidance and therefore influences ideal path selection.

Candidate Trajectory	Distance Weight (d_i)	COLREG Weight (c_i)		
		Head-on	Overtake	Crossover
1	1	1	2	3
2	2	2	1	2
3	3	3	2	0

TABLE III: Integer Distance and COLREG Weights

The final weight aims to discourage paths that are close to the contact vessel's predicted motion as these potentially increase collision risk. The proximity of a CV's predicted position to a candidate trajectory (P_x) is defined in Eq. 19. A negative proximity value suggests that the CV is predicted to cross the candidate trajectory in a direction not expected by collision regulations and thus yields a heavy weight penalty as collision risk is high. For positive proximity values, a linear trend is followed whereby the larger the proximity value the lower the weight that a candidate trajectory carries. The assignment uses a factor of the AV's beam to generate the integer weights as shown in table .

$$P_x = y_{ct} - y_{pr} \tag{19}$$

Proximity (P_x)	Collision Risk Weight (r_i)
$P_x < 0$	10
$0 < P_x < B_{av}$	5
$B_{av} \leqslant P_x \leqslant 3B_{av}$	1
$P_x > 3B_{av}$	0

TABLE IV: Integer CV Proximity Weight. *Where B_{av} is the Autonomous Vessel Beam and P_x is the proximity.*

The optimal candidate trajectory is then selected by solving the simple optimisation problem defined in Eq. 20. This solution simply summates the three trajectory weights into one trajectory cost and selects the candidate trajectory with this lowest total cost. In any case where the optimisation problem provides two trajectories which carry the same cost, decision favours in the direction of caution, selecting the trajectory with the further distance to increase opportunity for avoidance. The weighting system whilst suitable at a developmental level would again benefit from an expert-based method to determine the optimal weights as the current definition has limitations and oversights.

$$T_w = min \sum_{i=1}^{n} (d_i + c_i + r_i) \tag{20}$$

The autonomous vessel follows the local path until the avoidance action is considered sufficient and it is safe to return to the global path and continue the journey. Once the CV has been tracked past a bearing associated with the aft side of the vessel, namely ($150 < \beta < 210$), avoidance can be considered complete.

IV. Results

The final focus of this paper is to evaluate the performance of the developed Guidance and Navigation Systems, with a focus on experimental testing. The Navigation System is evaluated first under experimental testing with a focus on the sub-tasks of perception and localisation. This provides an indication of stereovision sensor performance and suitability of the sensors for the application. Simulation testing provides a means of verification and validation for the independent Guidance branch. After which experimental collision avoidance performance of the combined Navigation and Guidance System is assessed.

Fig. 9: MTT Towing Tank - Experimental Environment

The test environment and vessels used in experimental testing can be seen in Fig. 9. The Grey Seabax that assumes the role of the autonomous vessel and The Tito Neri vessel shall assume the role of contact vessel. The test environment is further equipped with an OptiTrack© 3D tracking system as visualised in Fig. 10 that is used to extract ground truth vessel pose data for use during sensor evaluations. The perception and localisation tasks of the Navigation System run on-board the AV's Jetson TX2, whilst the remaining procedures run on the host PC (MacBook Pro A1278), with all communications being handled via ROS.

A. Perception

The perception solution was evaluated using experimental testing which focused upon the robustness of obstacle detection and the accuracy of positioning. Under the precision/recall KPIs defined in. 21 and 22, the obstacle detection procedure maintained high precision with increasing recall achieving a total precision of 97.4% and recall of 94.6%. However as the neural network was both trained and tested with one type of

obstacle in one environment, the significance of this result is limited to the specific experimental environment.

The accuracy of obstacle positioning was evaluated using the ground truth from an OptiTrack camera system providing sub-millimetre accuracy. Data was gathered within the ten metre range covered by the perception system, with a total of 280 data points being gathered across three tests with varied vessel positioning. The comparison of the ground truth and perception system provided a means for evaluation with a root mean square position error of 0.31 metres being found using the KPI defined Eq. 23.

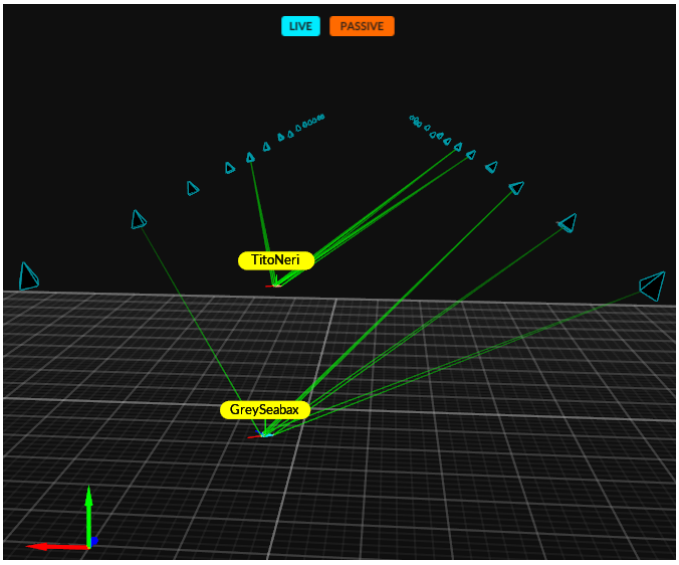


Fig. 10: OptiTrack© 3D Tracking System

B. Localisation

Experimental testing to evaluate localisation performance was conducted using two path types, a slalom/zigzag style path and a straight path. The two KPIs used are the position error and heading error, which are both defined by the root mean square deviance between the acquired sensor value and the accepted ground truth value (Eq. 24 and 25). Along the full 15m waterway length, the position error metric was found to be 0.129m and the heading error to be 5.3°. A variation in localisation performance was however demonstrated along the path length, with the performance decreasing with an increase in length. When the sampling length of the waterway is reduced to 7.5m, the position and heading errors also reduce significantly to 0.068m and 2.6° respectively. These evaluations were made under normal operational cases, however the localisation system was observed on multiple occasions to be susceptible to the impact of the kidnapped robot problem.

C. Collision Avoidance

The Guidance system was primarily subjected to simulation testing to verify and validate its collision avoidance procedure. To assess the COLREG procedure ten simulations were run for each of the three scenarios of head-on, overtake and crossover. The implemented collision avoidance procedure was found to succeed at avoiding collisions in all of these tests and do so in a manner compliant with collision regulation. The success of the collision avoidance procedure under simulation, allowed for testing to progress to the experimental phase, whereby the Guidance system relies upon the data from the navigation system.

Three interaction types were replicated during experimental testing, head-on, overtake and static. The crossover interaction type could not be replicated within the narrow test tank environment. Instead the ability of the AV to detect a static conflict vessel was tested as in this case it must assume full responsibility for avoidance. For each interaction type ten tests were conducted. Two performance metrics are used to evaluate the performance of the collision avoidance system, success rate and path optimality. The success rate provides an indication of a system's fundamental capability to avoid collision. Whereas the path optimality reviews how close the path selected by the avoidance system was to a most desirable selection. A hierarchy exists between the former and latter metric as the path optimality is dependent upon a successful avoidance case for the assignment of an optimality value.

The success rate in this application is quantified as a percentage and is divided by three outcomes, compliant, semi-compliant and failed. Compliant collision avoidance indicates that the AV has successfully avoided collision, following the expected protocol. Semi-compliant collision avoidance indicates that collision avoidance was successful but the expected protocol was not adhered to exactly as expected. Finally, failed avoidance indicates that the AV was unsuccessful in avoiding a collision. The path optimality metric is defined in Eq. 26. It describes the relation between the distance the local avoidance path added to the global route and the collision risk, quantified by the shortest distance between the AV and CV during the interaction. A lower metric value indicates optimality.

Scenario Number	**Collision Avoidance**			**Optimality Metric**
	Compliant	**Semi-compliant**	**Failed**	
Static	40%	50%	10%	0.4
Head-on	30%	50%	20%	0.57
Overtake	50%	50%	0%	0.33
Average	**40%**	**50%**	**10%**	**0.43**
Simulation	*100%*	*0%*	*0%*	*0.38*

TABLE V: Collision Avoidance Evaluation. *Compliant: successful avoidance as per procedure, semi-compliant: successful avoidance however not exactly per procedure, Failed: unsuccessful avoidance. Metric: path optimality KPI.*

Experimental collision avoidance results can be seen in Table V. Across all of the thirty tests conducted, an average avoidance success rate of 90% was achieved, meaning that 10% of interactions resulted in collision due to insufficient action. Of the successful tests however, only 40 of the 90% percent saw fully compliant avoidance with the remainder seeing either the premature activation of emergency procedure or the incorrect COLREG mode being activated. This behaviour is reflected in the resultant path optimality, with the mean average being 0.43, a reduction in performance of that through simulation.

The experimental testing has highlighted that whilst the Guidance System performed particularly well under simulation, with real world data supplied from the stereovision devices this same performance could not be accomplished. Despite almost always generating a path that successfully avoided collision with a contact vessel, experimental avoidance was not capable of consistently triggering the correct response due to the data from the navigation system not being reliable enough.

V. Conclusions

A. Recapitulation

A stereovision based navigation system, with a multi-device system has been proposed to achieve mid-range perception and localisation onboard inland autonomous vessels. A multi-device perception setup has been introduced that enables the autonomous vessel to maintain a constant lookout in areas of collision risk. The utilisation of a convolutional neural network to conduct object detection has demonstrated its suitability to the autonomous inland vessel application. A refined approach was taken towards the processing of depth data which avoids the computationally expensive manipulation of dense point clouds, by directly locating the obstacles closest point of approach. A dedicated stereovision tracking camera, utilising a Visual Inertial Odometry approach was also integrated to provide pose data for autonomous vessel localisation.

The perception solution was evaluated using experimental testing which focused upon the robustness of obstacle detection and the accuracy of positioning. The obstacle detection procedure was found to run at a high inference rate of 25ms on the onboard processing unit, whilst maintaining high precision with increasing recall. However as the neural network was both trained and tested with one type of obstacle in one environment, the significance of this result is limited to the specific experimental environment. The accuracy of obstacle positioning was also evaluated and the perception system exhibited an RMS position error of 0.31 metres.

Experimental testing to evaluate localisation performance was conducted using two path types, a slalom/zigzag style path and a straight path. Along a 15m length of waterway, the position error metric was found to be 0.129m and the heading error to be 5.3°. The localisation performance was seen to reduce as the vessel progressed down the path length and the system was observed on multiple occasions to be susceptible to the impact of the kidnapped robot problem, limiting its reliability slightly.

The developed Guidance System incorporates a collision avoidance protocol capable of handling the specific challenges of the inland waterway. A new configuration for conflict detection regions has been proposed to overcome the issues that arise from the inland waterway architecture and the subsequent nature of the interactions that take place. The collision resolution approach integrates collision regulations and a secondary emergency procedure within a rule-based implementation compatible with the data available from the stereovision based Navigation System. The local path planner incorporates a roll-out-trajectory generation technique that has been adapted and optimised for application on an inland autonomous vessel.

Collision avoidance experiments provided an evaluation of the combined guidance and navigation system performance. Across all of the thirty tests conducted, an average avoidance success rate of 90% was achieved, meaning that 10% of interactions resulted in collision due to insufficient action. Of the successful tests however, only 40 out of the total 90% percent saw fully compliant avoidance with the remainder seeing either the premature activation of emergency procedure or the incorrect COLREG mode being activated.

The experimental testing has highlighted that whilst the Guidance System exceptionally well under simulation, with real world data supplied from the stereovision devices this same performance could not be accomplished. Despite almost always generating a path that successfully avoided collision with a contact vessel, experimental avoidance was not capable of consistently triggering the correct response due to the data from the navigation system not being dependable enough. Specifically, the weakest link in the entire network during this combined testing appeared to be the reliability of the localisation system.

B. Future Work

A focus on inland data acquisition for the training and testing of CNN based obstacle detection would be an enabler to future research. Gathering image data on inland waterway obstacles and developing an Inland Maritime Dataset would enable research to apply focus to model optimisation, perhaps utilising a similar initiative to the KITTI benchmark suite [29].

Further development of the Navigation System with a view to the most appropriate approach for complementing the stereovision setup through fusion with data from other sensor sets. GPS could be aptly coupled with the stereovision sensors for localisation, providing global re-localisation along the path length. For perception, the stereovision solution would benefit from coupling with a radar sensor to extend range and ultrasonic sensors to cover the area around the vessel.

Regarding the collision avoidance procedure, a primary focus should be towards the handling of avoidance in congested waterways with multiple contact vessels present at one time, expanding upon the current approach. Secondary research could focus towards a more advanced approach to motion prediction, considering vessel behaviour on the inland waterway. The definitions of conflict region dimensions and path weighting would further benefit from testing and fine tuning along varied waterway stretches.

REFERENCES

1 van Riessen, B., Negenborn, R. R., and Dekker, R., "Synchromodal container transportation: An overview of current topics and research opportunities," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9335, pp. 386–397, 2015.

2 Peeters, G., Kotzé, M., Afzal, M. R., Catoor, T., Van Baelen, S., Geenen, P., Vanierschot, M., Boonen, R., and Slaets, P., "An unmanned inland cargo vessel: Design, build, and experiments," *Ocean Engineering*, vol. 201, 2020.

3 Wang, W., Gheneti, B., Mateos, L. A., Duarte, F., Ratti, C., and Rus, D., "Roboat: An Autonomous Surface Vehicle for Urban Waterways," pp. 6340–6347, 2020.

4 Song, L., Chen, H., Xiong, W., Dong, Z., Mao, P., Xiang, Z., and Hu, K., "Method of Emergency Collision Avoidance for Unmanned Surface Vehicle (USV) Based on Motion Ability Database," *Polish Maritime Research*, vol. 26, no. 2, pp. 55–67, 2019.

5 Shi, B., Su, Y., Wang, C., Wan, L., and Luo, Y., "Study on intelligent collision avoidance and recovery path planning system for the waterjet-propelled unmanned surface vehicle," *Ocean Engineering*, vol. 182, no. May, pp. 489–498, 2019. [Online]. Available: https://doi.org/10.1016/j.oceaneng.2019.04.076

6 Song, L., Chen, Z., Dong, Z., Xiang, Z., Mao, Y., Su, Y., and Hu, K., "Collision avoidance planning for unmanned surface vehicle based on eccentric expansion," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, pp. 1–9, 2019.

7 Sun, X., Wang, G., Fan, Y., Mu, D., and Qiu, B., "Fast Collision Avoidance Method Based on Velocity Resolution for Unmanned Surface Vehicle," *Proceedings of the 31st Chinese Control and Decision Conference, CCDC 2019*, pp. 4822–4827, 2019.

8 Fan, Y., Sun, X., and Wang, G., "An autonomous dynamic collision avoidance control method for unmanned surface vehicle in unknown ocean environment," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, pp. 1–11, 2019.

9 yuan Zhuang, J., Zhang, L., qi Zhao, S., Cao, J., Wang, B., and bing Sun, H., "Radar-based collision avoidance for unmanned surface vehicles," *China Ocean Engineering*, vol. 30, no. 6, pp. 867–883, 2016.

10 Lyu, H. and Yin, Y., "COLREGS-Constrained Real-Time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields," *Journal of Navigation*, vol. 72, no. 3, pp. 588–608, 2019.

11 Eriksen, B. O. H., Wilthil, E. F., Flåten, A. L., Brekke, E. F., and Breivik, M., "Radar-based maritime collision avoidance using dynamic window," *IEEE Aerospace Conference Proceedings*, vol. 2018-March, no. March, pp. 1–9, 2018.

12 Wang, Y., Chao, W. L., Garg, D., Hariharan, B., Campbell, M., and Weinberger, K. Q., "Pseudo-lidar from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 8437–8445, 2019.

13 Huang, Y., Chen, L., Chen, P., Negenborn, R. R., and van Gelder, P. H., "Ship collision avoidance methods: State-of-the-art," *Safety Science*, vol. 121, no. April 2019, pp. 451–473, 2020.

14 Lee, S. M., Kwon, K. Y., and Joh, J., "A fuzzy logic for autonomous navigation of marine vehicles satisfying COLREG guidelines," *International Journal of Control, Automation and Systems*, vol. 2, no. 2, pp. 171–181, 2004.

15 Tan, G., Zou, J., Zhuang, J., Wan, L., Sun, H., and Sun, Z., "Fast marching square method based intelligent navigation of the unmanned surface vehicle swarm in restricted waters," *Applied Ocean Research*, vol. 95, no. December 2019, p. 102018, 2020. [Online]. Available: https://doi.org/10.1016/j.apor.2019.102018

16 Eriksen, B.-O. H., Breivik, M., Wilthil, E. F., Flåten, A. L., and Brekke, E. F., "The Branching-Course MPC Algorithm for Maritime Collision Avoidance."

17 Kotzé, M., Junaid, A. B., Afzal, M. R., Peeters, G., and Slaets, P., "Use of Uncertainty Zones for Vessel Operation in Inland Waterways," *Journal of Physics: Conference Series*, vol. 1357, no. 1, 2019.

18 Singh, Y., Sharma, S., Hatton, D., and Sutton, R., "Optimal path planning of unmanned surface vehicles," *Indian Journal of Geo-Marine Sciences*, vol. 47, no. 7, pp. 1325–1334, 2018.

19 Li, S., Liu, J., Negenborn, R. R., and Ma, F., "Optimizing the joint collision avoidance operations of multiple ships from an overall perspective," *Ocean Engineering*, vol. 191, no. May, p. 106511, 2019. [Online]. Available: https://doi.org/10.1016/j.oceaneng.2019.106511

20 Łącki, M., "Neuroevolutionary approach to colregs ship maneuvers," *TransNav*, vol. 13, no. 4, pp. 745–750, 2019.

21 Vallestad, I. J., "Path Following and Collision Avoidance for Marine Vessels with Deep Reinforcement Learning," no. June, 2019.

22 Namgung, H., Jeong, J. S., Kim, J. S., and Kim, K. I., "Inference Model of Collision Risk Index based on Artificial Neural Network using Ship Near-Collision Data," *Journal of Physics: Conference Series*, vol. 1357, no. 1, 2019.

23 Darweesh, H., Takeuchi, E., Takeda, K., Ninomiya, Y., Sujiwo, A., Morales, L. Y., Akai, N., Tomizawa, T., and Kato, S., "Open source integrated planner for autonomous navigation in highly dynamic environments," *Journal of Robotics and Mechatronics*, vol. 29, no. 4, pp. 668–684, 2017.

24 Benjamin, M. R., Leonard, J. J., Curcio, J. A., and Newman, P., "A method for protocol-based collision avoidance between autonomous marine surface craft," *J. Field Robotics*, vol. 23, pp. 333–346, 2006.

25 Prasad, D. K., Rajan, D., Rachmawati, L., Rajabally, E., and Quek, C., "Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 1993–2016, 2017.

26 Shao, Z., Wu, W., Wang, Z., Du, W., and Li, C., "Seaships: A large-scale precisely annotated dataset for ship detection," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2593–2604, 2018.

27 Kim, K., Hong, S., Choi, B., and Kim, E., "Probabilistic ship detection and classification using deep learning," *Applied Sciences*, vol. 8, no. 6, p. 936, Jun 2018. [Online]. Available: http://dx.doi.org/10.3390/app8060936

28 Fossen, T. I., *Handbook of Marine Craft Hydrodynamics and Motion Control*, 2011.

29 Geiger, A., Lenz, P., and Urtasun, R., "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

APPENDIX I - KEY PERFORMANCE INDICATORS

A. Perception

Precision and recall are defined as shown in Equations 21 and 22 respectively.

$$Precision = \frac{correct\ predictions}{all\ predictions} \tag{21}$$

$$Recall = \frac{correct\ detections}{all\ ground\ truths} \tag{22}$$

The definition of obstacle position error is given in Eq. 23 below with $\hat{o_i}$ referring to the ground truth position of the obstacle, o_i being the measured position value and N being the number of samples.

$$RMSE_{pos} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{o_i} - o_i)^2}{N}} \tag{23}$$

B. Localisation

The definitions the two Absolute Trajectory Error metrics can be defined below in Eq. 24 and 25 relating to the position and rotation error respectively with $\hat{p_i}$ and $\hat{y_i}$ being the ground truth values, p_i and y_i being the measurement values and N being the number of samples.

$$ATE_{pos} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{p_i} - p_i)^2}{N}} \tag{24}$$

$$ATE_{rot} = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y_i} - y_i)^2}{N}} \tag{25}$$

C. Collision Avoidance

$$Path\ Optimality \quad = \quad \frac{Additional\ Path\ Distance\ (m)}{Collision\ Vicinity\ (m)} \tag{26}$$

APPENDIX II - ALGORITHMS

Algorithm 1: Obstacle Replication Removal

```
Result: Replication Removal
for i Fore_obstacles do
    for j in Port_obstacles and Stbd_obstacles do
        Vicinity = |coords[i] − coords[j]|
        if Vicinity ⩽ 0.5m and
         classification[i]==classification[j] then
            Remove(j)
    end
end
```

Algorithm 2: Collision Regulation Procedure

```
Result: COLREG Status and Avoidance
if u_cv = 0 and not 30 < β < 330 then
    status = 'GiveWay'
    avoidance = 'Overtake'
else if 90 < α < 270 then
    if 0 < β < 90 and α > 195 then
        status = 'GiveWay'
        avoidance = 'Crossover'
    else if β ⩾ 350 and α ⩾ 165 then
        status = 'GiveWay'
        avoidance = 'HeadOn'
    else if β ⩽ 10 and α ⩾ 165 then
        status = 'GiveWay'
        avoidance = 'HeadOn'
    else
        status = 'StandOn'
        avoidance = 'Standby'
    end
else
    if < β < 112.5 and 270 < α < 345 then
        status = 'GiveWay'
        avoidance = 'Crossover'
    else if u_cv < u_av and not 30 < β < 330 then
        if β > 330 and α ⩽ 15 then
            status = 'GiveWay'
            avoidance = 'Overtake'
        else if β < 30 then
            status = 'GiveWay'
            avoidance = 'Overtake'
        else
            status = 'StandOn'
            avoidance = 'Standby'
        end
    else
        status = 'StandOn'
        avoidance = 'Standby'
    end
end
```

Algorithm 3: Emergency Avoidance Procedure

```
Result: Emergency Avoidance
status = 'Emergency'
if u_cv = 0 then
    if not 5 < β < 355 then
        avoidance = 'EmergencyStop'
    else if β > 180 then
        avoidance = 'HardStarboard'
    else
        avoidance = 'HardPort'
    end
else if β > 315 then
    if α < 165 then
        avoidance = 'HardPort'
    else
        avoidance = 'HardStarboard'
    end
else if β < 45 then
    if α > 195 then
        avoidance = 'HardStarboard'
    else
        avoidance = 'HardPort'
    end
else
    avoidance = 'CollisionImminent'
end
```

B

Appendix B - Simulation Results

Simulations results for scenarios where the AV was required to take action can be found in the tables below. In each test both vessels were defined a start position and state that should yield the relevant conflict to be detected and a new local path to be generated. Following the detection of conflict, over a horizon of twenty time steps (seconds), it is assumed that the AV follows the generated path exactly and the CV is anticipated to act accordingly with regulations through remaining on its path or in the case of the head-on scenario, also take action to avoid collision. For this prediction of motion it is assumed that the CV takes action identical to that of the AV.

Test Number	**Collision Avoidance**			**Path Optimality Metric**
	Compliant	**Semi-compliant**	**Failed**	
1	✓			0.1
2	✓			0.1
3	✓			0.11
4	✓			0.13
5	✓			0.77
6	✓			0.77
7	✓			0.51
8	✓			0.5
9	✓			0.51
10	✓			0.13
Average	100%	0 %	0%	0.33

Table B.1: Simulation Results from Head-On Scenario.

Test Number	Collision Avoidance			Path Optimality Metric
	Compliant	Semi-compliant	Failed	
1	✓			0.51
2	✓			0.34
3	✓			0.60
4	✓			0.43
5	✓			0.38
6	✓			0.43
7	✓			0.40
8	✓			0.3
9	✓			0.33
10	✓			0.43
Average	100%	0 %	0%	0.42

Table B.2: Simulation Results from Overtake Scenario.

Test Number	Collision Avoidance			Path Optimality Metric
	Compliant	Semi-compliant	Failed	
1	✓			0.31
2	✓			0.42
3	✓			0.39
4	✓			0.36
5	✓			0.51
6	✓			0.58
7	✓			0.66
8	✓			0.31
9	✓			0.29
10	✓			0.19
Average	100%	0 %	0%	0.40

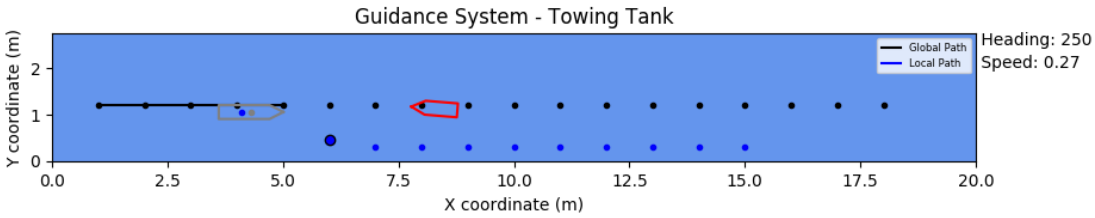
Table B.3: Simulation Results from Crossover Scenario.

Test Number	Collision Avoidance			Path Optimality Metric
	Compliant	Semi-compliant	Failed	
1	✓			1.82
2	✓			0.39
3	✓			0.77
4	✓			0.70
5	✓			0.61
6	✓			0.70
7	✓			0.36
8	✓			0.58
9	✓			1.52
10	✓			0.36
Average	100%	0 %	0%	0.78

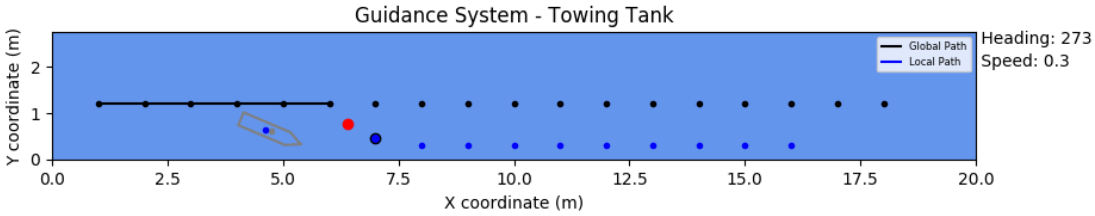
Table B.4: Simulation Results from Emergency Scenario.

C

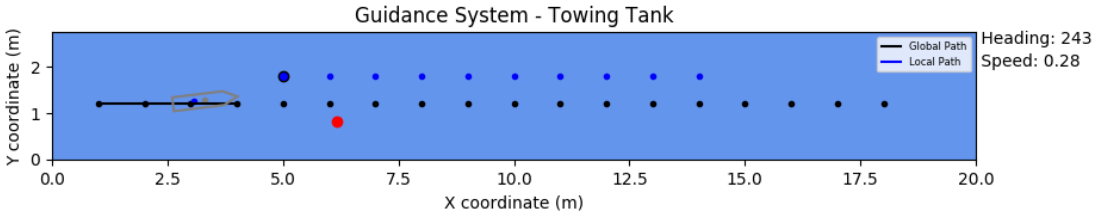
Appendix C - Experimental Results



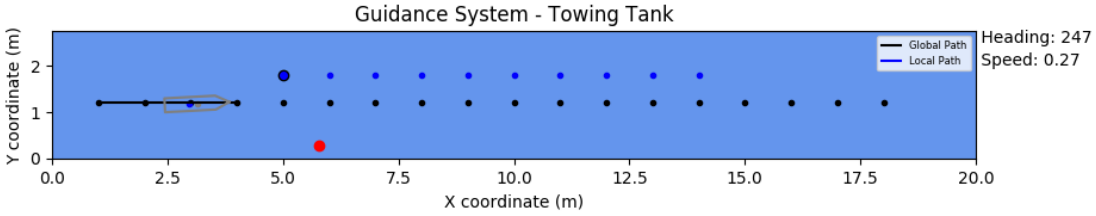
(a) Static Test 1



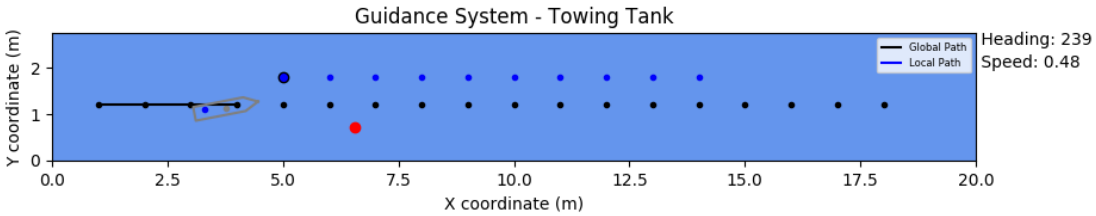
(b) Static Test 2



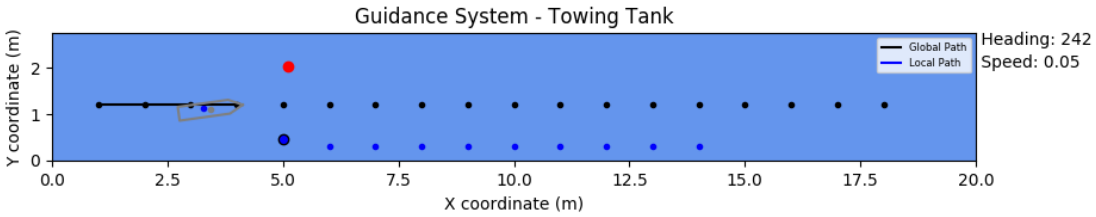
(c) Static Test 3



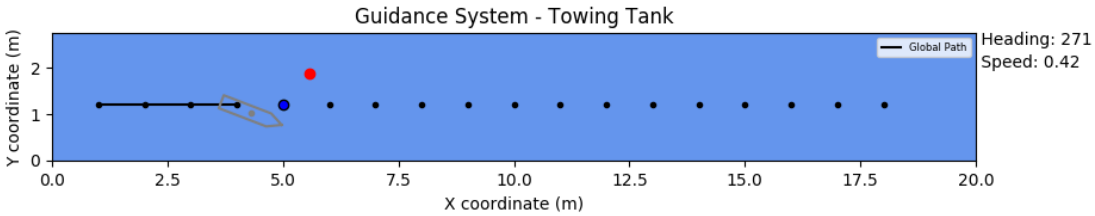
(d) Static Test 4



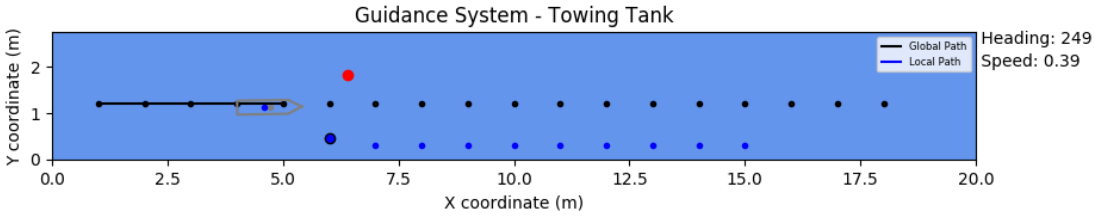
(e) Static Test 5



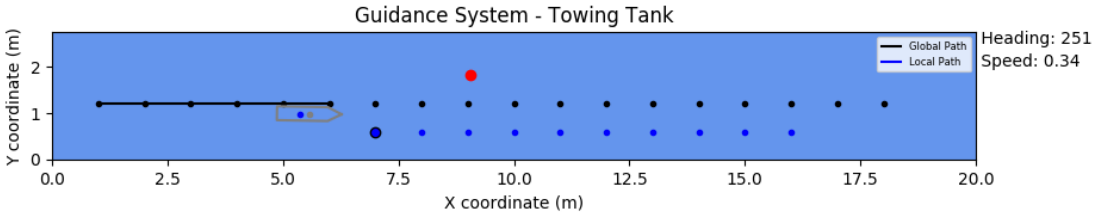
(f) Static Test 6



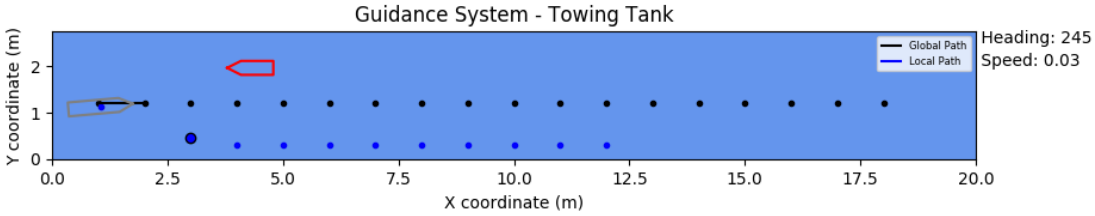
(g) Static Test 7



(h) Static Test 8

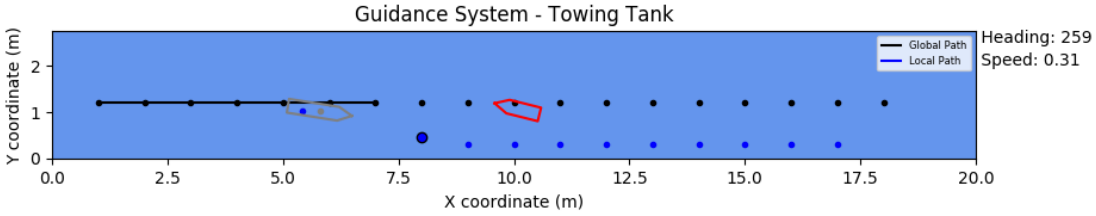


(i) Static Test 9

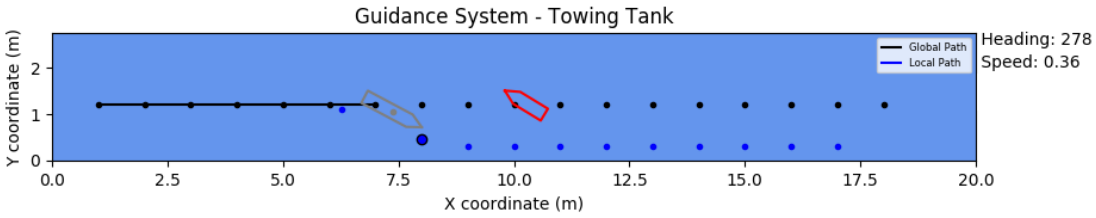


(j) Static Test 10

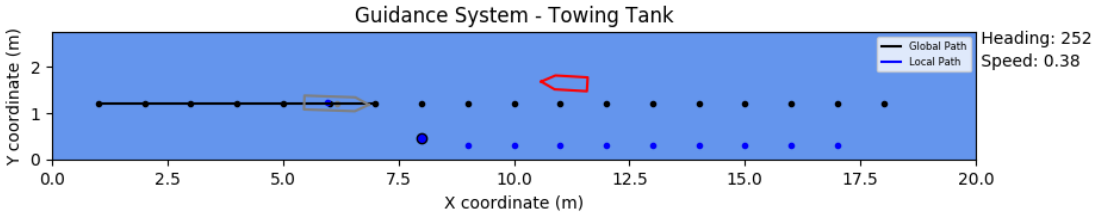
Figure C.1: Results of Experimental Collision Avoidance- Static Contact Vessel Scenario



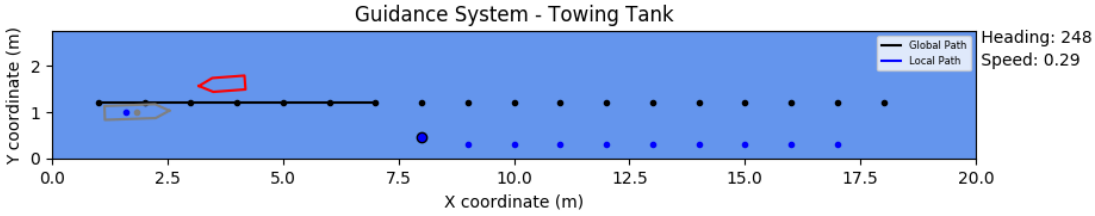
(a) Head On Test 1



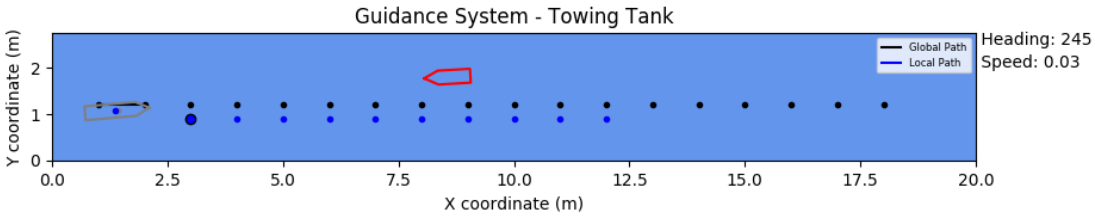
(b) Head On Test 2



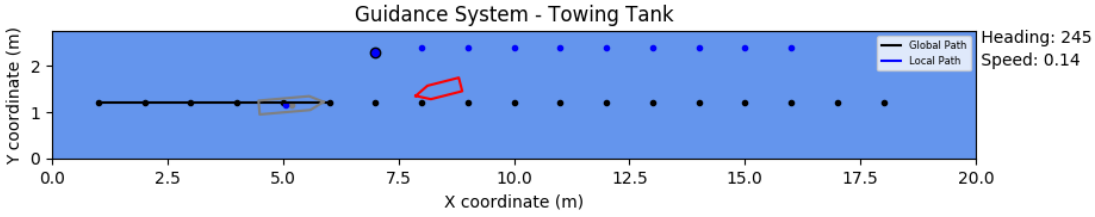
(c) Head On Test 3



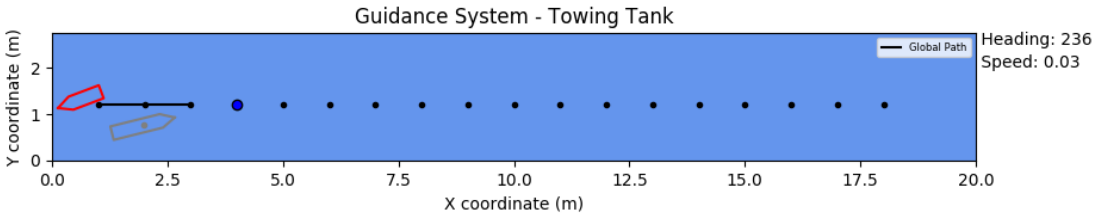
(d) Head On Test 4



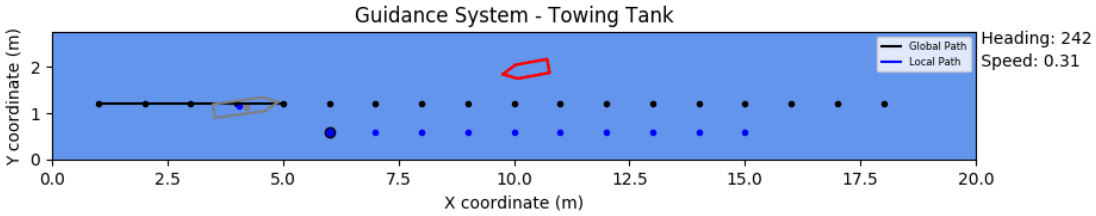
(e) Head On Test 5



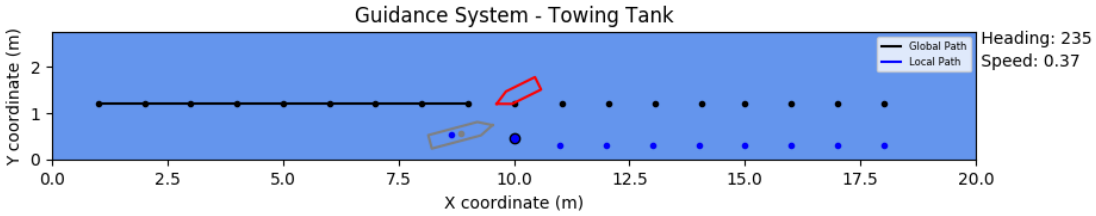
(f) Head On Test 6



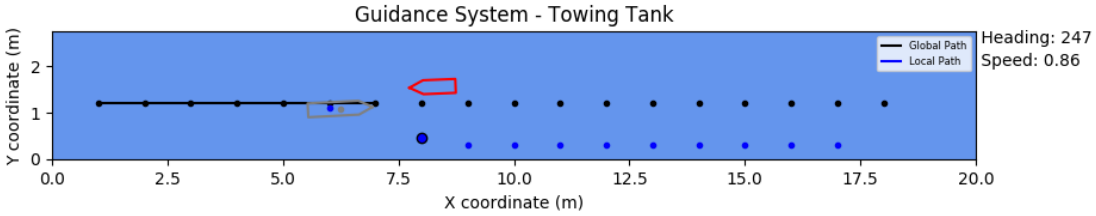
(g) Head On Test 7



(h) Head On Test 8

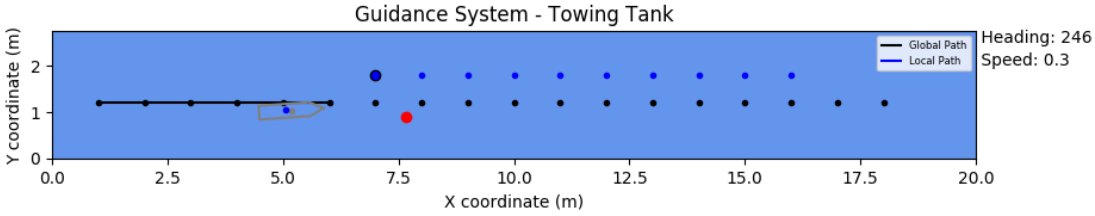


(i) Head On Test 9

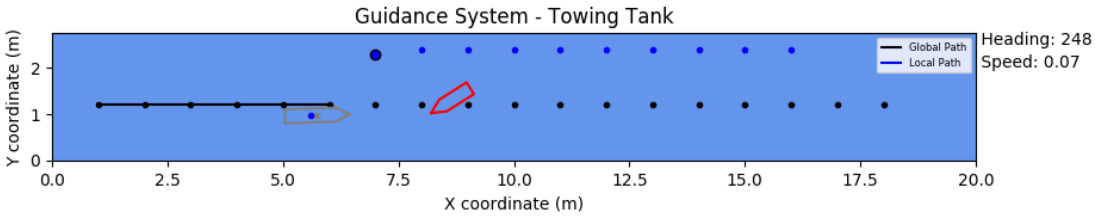


(j) Head On Test 10

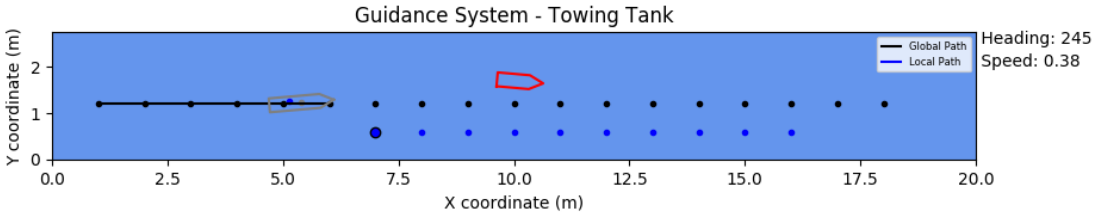
Figure C.2: Results of Experimental Collision Avoidance- Head On Scenario



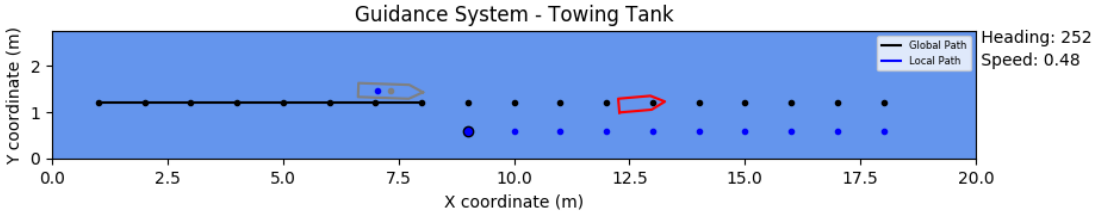
(a) Overtake Test 1



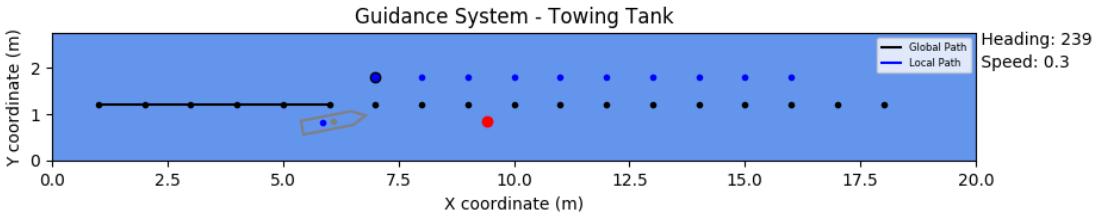
(b) Overtake Test 2



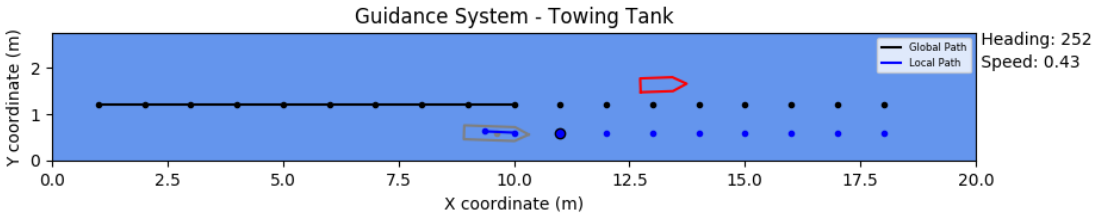
(c) Overtake Test 3



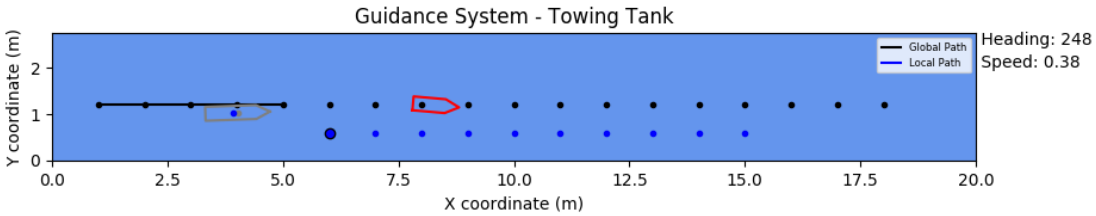
(d) Overtake Test 4



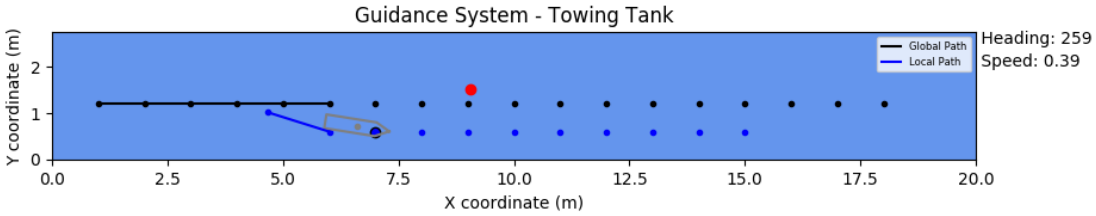
(e) Overtake Test 5



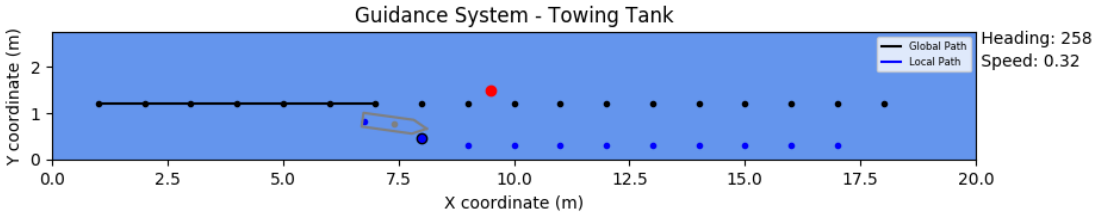
(f) Overtake Test 6



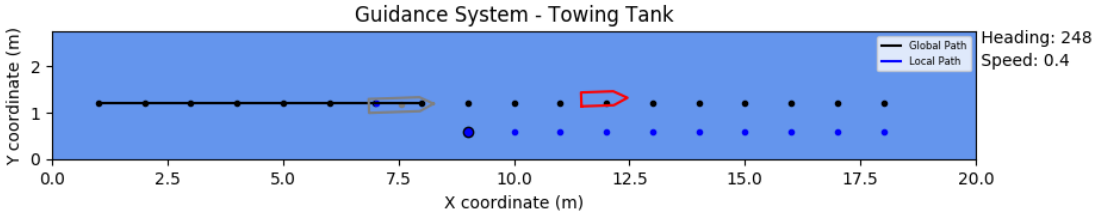
(g) Overtake Test 7



(h) Overtake Test 8



(i) Overtake Test 9



(j) Overtake Test 10

Figure C.3: Results of Experimental Collision Avoidance- Overtake Scenario