Negative Obstacle Detection and Localisation: A Combined Deep Learning and Computer Vision Framework

Jihad Dib

School of Engineering

University of Kent

A Thesis Submitted to the University of Kent for the Degree of

Doctor of Philosophy in Electronic Engineering

May 2024

In the name of Allah, the Most Gracious, the Most Merciful.

"Read in the name of your Lord Who created, created man from a clinging clot.

Read, and your Lord is the Most Generous, who taught by the pen, taught man what he did not know." (Al-'Alaq 96:1-5)

Acknowledgements

I would like to express my sincere appreciation to my supervisors, Dr. Sanaul Hoque, Prof. Konstantinos Sirlantzis, and Prof. Gareth Howells, for their unwavering encouragement and invaluable guidance and support. Their expertise and willingness to share their priceless knowledge have significantly enriched my research experience. I am forever indebted to them, and their words of wisdom will never be forgotten.

I would also like to thank every member of staff within the School of Engineering at the University of Kent for their logistical and administrative support in overcoming all the operational obstacles I faced throughout this journey.

I would like to gift this work to my beloved princess, my newborn daughter, Talia, God's Gift with whom my life has been blessed, and to my life partner, my soulmate, my wife Ibtisam, who supported me throughout this journey. Thank you for believing in me, and thank you for your endless encouragement in all its forms.

I would also like to gift this work and express my heartfelt gratitude, appreciation, and thankfulness to my parents, Abed Jawad Dib and Jinan Ammar. No words can describe how thankful I am to you. I would not be who I am if it weren't for your never-ending love, support, and encouragement. You were my backbone throughout my life, and your prayers shaped my life. I would also like to thank my younger brother, Jad.

I extend my thankfulness to my colleagues within the Intelligent Interactions Team, Sotirios Chatzidimitriadis, Saber Mirzaee Bafti, Odysseas Doumas, Elhassan Mohamed, and Yankun Yang. We crossed the same path together; the times and experiences we shared will always be remembered.

Project ABSAR 'AI-Based Assistive Robotics' is supported by the Engineering and Physical Sciences Research Council (EPSRC) National Productivity Investment Fund via studentship reference 2112938.

List of Publications

1. E. Mohamed, J. Dib, K. Sirlantzis and G. Howells, “Integrating ride dynamics measurements and user comfort assessment to smart robotic wheelchairs,” 2019 15th Conference on Global Challenges in Assistive Technology: Research, Policy & Practice, Bologna, Italy/ IOS Press. Available at: https://aaate2019.eu/aaate-2019-proceedings/.
2. J. Dib, K. Sirlantzis and G. Howells, "A Review on Negative Road Anomaly Detection Methods," in IEEE Access, vol. 8, pp. 57298-57316, 2020, doi: 10.1109/ACCESS.2020.2982220.
3. E. Mohamed, K. Sirlantzis, G. Howells and J. Dib, "Investigation of Vibration and User Comfort for Powered Wheelchairs," in IEEE Sensors Letters, vol. 6, no. 2, pp. 1-4, Feb. 2022, Art no. 2500204, doi: 10.1109/LSENS.2022.3147740
4. Dib, J., Sirlantzis, K. and Howells, G. (2022). Application of Deep Learning Techniques in Negative Road Anomalies Detection. In Proceedings of the 14th International Joint Conference on Computational Intelligence – ROBOVIS, Valetta, Malta; ISBN 978-989-758-611-8; ISSN 2184-3236, SciTePress, pages 475-482. DOI: 10.5220/0011336000003332
5. Dib, J., Sirlantzis, K. and Howells, G. (2023). “An annotated water-filled dry potholes dataset for deep learning applications”. Data in Brief, Volume 48(109206), p.109206. doi:https://doi.org/10.1016/j.dib.2023.109206.
6. Dib, J., Sirlantzis, K., Howells, G., Hoque S. “A Novel Real-Time Negative Obstacles Detection and Localisation Technique” (Under Review)
7. Dib, J., Sirlantzis, K., Howells, G., Hoque S. “Real-Time Safe Traversal of Inclines, Ramps, and Paths via 3D Point Cloud Segmentation“ (Under Review)

Abstract

The burgeoning advancements in robotics and Artificial Intelligence (AI) have propelled a pervasive application in diverse domains, paving the path towards countless different applications ranging from path planning and object detection to autonomous and semi-autonomous navigation. However, for autonomy in navigation to be achievable, robotic platforms are expected to be able to traverse different types of thoroughfares. For this to be achieved, systems should be able to take into account different obstacles that are needed and localised in order for them to be avoided. One of the main platforms being widely and heavily used are assistive technologies such as mobility scooters, segways, and electric-powered wheelchairs (EPWs). Recent advancements in technology have made positive obstacle avoidance possible and very accurate, as there have been many stable and highly reliable systems with the help of different technologies ranging from multimodal sensing techniques to computer vision. However, one of the main challenges which persist is negative road anomalies, upward and downward-inclined paths, and curbs when it comes to assistive technologies.

For these systems to be reliable, novel techniques and approaches are a must due to the fact that assistive technologies are universally considered safety-critical systems, i.e. systems that could directly impact human safety, whereby failure could potentially lead to serious injury or death.

This thesis introduces three different approaches aimed at tackling various problems that render autonomous and semi-autonomous navigation possible. The first problem is Negative Road Anomalies, depressions, or irregular paths and roads caused by wear and tear within the ground surface, including different types of pavement and road imperfections. One of the most universal and fundamental examples of road anomalies is potholes, or road depressions, which form the highest threat to assistive technologies. As a solution to this problem, this thesis proposes a novel object detection and localisation algorithm based on deep learning and machine vision. The second problem addressed within this thesis is upward and downward path inclinations such as ramps and dropped

curbs. This thesis proposes a multimodal sensing technique that uses stereo vision and depth vision along with an inertial measurement unit (IMU) in order to detect, localise, and assess inclined planes by processing the point cloud generated as a result of the mentioned sensing techniques. The proposed approach can detect, segment, and localise the inclined plane, whether it is upward-facing or downward-facing. It can also calculate the inclination angle and any ground offset, as well as estimate the width of the inclined plane in order to assess whether it is traversable or not.

As for the third problem addressed within this thesis, it is edge detection, localisation, and avoidance, in order to avoid the risk of falls. This enables assistive technologies to detect the edge of the road whether it is a cliffside, curb, or any physical end of a road or path. This can be achieved by using the same sensing techniques proposed for inclined planes traversal, but with a completely different assessment logic.

The proposed systems were tested in a real-life scenario by mounting them onto a widely and heavily used assistive robotic system from the realm of assistive mobility, an Electric-Powered Wheelchair, or EPW. The three proposed systems were mounted onto an EPW, which was driven in real-life conditions indoors and outdoors as a representative of everyday user contexts as well as in isolated and controlled laboratory settings for in-depth performance assessment of reliability and accuracy. Due to the critical nature of the systems, the ground truth has been measured with the use of indisputable techniques such as a tape measure and a protractor in order to safely assess the system's performance to the millimetre and decimal degree accuracy.

Table of Contents

1. **Introduction** **27**

1.1 Research Motivation 29

1.2 Aim and Objectives 31

1.3 Research Questions 31

1.4 Main Contributions 33

1.5 Applications 33

1.6 Scope of the Thesis 34

1.7 Structure of the Thesis 36

2. A Review on Negative Obstacles and Inclines Detection **40**

2.1 Introduction 40

2.2 Assistive Technologies, the Motivation 42

2.3 Negative Road Anomalies Detection Techniques 44

2.3.1 Review Strategy and Performance Measures 45
2.3.2 Vision-Based Systems 46
2.3.3 Non-Vision Based Systems 63

2.4 Incline Detection and Assessment Techniques 68

2.4.1 Review Strategy and Performance Measures 69
2.4.2 Vision or Radar-Based Systems 70
2.4.3 Non-Vision or Radar-Based Systems 76

2.5 Downwards-facing Stairs and Curbs Detection and Localisation Techniques 80

2.5.1 Review Strategy and Performance Measures 81
2.5.2 Vision or Radar-Based Systems 83
2.5.3 Non-Vision or Radar-Based Systems 88

2.6 Discussion and Conclusion 90

3. A Manually Collected Potholes Images Dataset 94

3.1 Introduction 94

3.2 Statement of Problem 95

3.3 Data Collection 96

3.4 Motivation 96

3.5 The Manually-Collected Dataset 98

3.6 Conclusion 103

4. Negative Road Anomalies Detection and Localisation 106

4.1 Introduction 106

4.2 Methodology 108

4.2.1 Statement of Problem 109

4.2.2 System Architecture 110

4.3 Experiment 119

4.4 Results and Discussion 122

4.4.1 Network Training 122

4.4.2 Spatial Understanding of the Environment 133

4.5 System's Limitations 139

4.6 Conclusion 140

5. Incline Detection and Localisation via Point Cloud Imaging 143

5.1 Introduction 143

5.2 Methodology 145

5.2.1 Statement of Problem 145

5.2.2 System Architecture 145
5.3 Experiment 153
5.4 Results and Discussion 156
5.5 System's Limitations 166
5.6 Conclusion 167

6. Curb and Downward-facing stairs Detection and Localisation 170

6.1 Introduction 170
6.2 Methodology 173
6.2.1 Statement of Problem 173
6.2.2 System Architecture 174
6.3 Experiment 179
6.4 System's Limitations 193
6.5 Conclusion 194

7. Conclusion 196

7.1 Concluding Comments 196
7.2 Main Contributions and Research Findings 197
7.3 Future Work Recommendations 204

8. References 207

List of Figures

3.1 **Sample images are taken from the manually-collected dataset** 97

3.2 **Distribution of potholes in the proposed Dataset** 99

3.3 **Number of potholes in images** 99

3.4 **Image labelling as completed using the open-source software described in section** 100

3.5 **YOLO Darknet Annotation Format** 101

3.6 **Pascal VOC XML Annotation Format** 102

3.7 **Supplemental dataset samples** 104

4.1 **Top-Level System Architecture** 110

4.2 **Diagram representing the first layer of the system** 112

4.3 **ResNet residual block** 114

4.4 **Tiny YOLO Network Architecture** 116

4.5 **Pothole as seen by the RGB-D Camera** 117

4.6 **Location of the imaging sensor within the Intel D435i Camera** 118

4.7 **Depth calculation start point** 118

4.8 **The four main points of interest (POIs), in addition to the centre point** 119

4.9 **RGB-D Camera mounted onto the EPW** 121

4.10 **Experiment location and boundary** 121

4.11 **DenseNet 201 training resulting graph** 124

4.12 **ResNet50 training resulting in graph** 125

4.13 **Tiny YOLO training resulting in graph** 126

4.14 **Video comparison of the three networks** 129

4.15 **Tiny YOLO v4 Confidence Rate** 130

4.16 **Real-time testing of the YOLO Network.** 132

4.17 **Error distribution of the system's detection rate** 138

5.1 **Segmented Plane** 147

5.2 **Width of the Electric-Powered Wheelchair** 149

5.3 **A segmented plane(green), its concave hull (in red), and the estimated width of the plane (in blue)** 150

5.4 **The plane as observed by the RGB-D Camera (left) and its resulting segmented plane (right)** 150

5.5 **Wheelchair ramp as seen by the Intel RealSense D435i camera sensors** 151

5.6 **The wheelchair ramp from Figure 5.4 after segmentation and assessment.** 152

5.7 **Experiment setup** 154

5.8 **Experiment setup.** 154

5.9 **Isolated environment experiment** 155

5.10 **Average angle estimation error** 159

5.11 **"Rutherford" ramp as seen by the Intel RealSense D435i camera** 160

5.12 **"Rutherfordrampup" segmented (left) with its inclination and width estimated by the proposed algorithm (right)** 161

5.13 **Stiff bicycle ramp as seen by the Intel RealSense D435i Camera** 161

5.14 **"Reject 1" test results as returned by the proposed algorithm** 162

5.15 **"Reject 1" ground truth as measured by "Clinometer"** 162

5.16 **"Lab 2" experiment instance within the indoors scenario as represented in Table 5.2** 163

5.18 **"Lab 2" experiment instance output returned by the proposed algorithm** 163

5.19 **Second experiment's "Lab 16" instance inclination angle as measured by the protractor.** 164

5.20 **"Lab 16" instance as measured by "Clinometer"** 164

5.21 **"Lab 16" Instance experiment result as returned by the proposed algorithm** 164

6.1 **Centroid A, Zmin, Zman, and mD (minimumSafeDistance)** 177

6.2 **Traversable area** 178

6.3 **Curb scenario as seen by the RealSense D435i camera** 179

6.4 **Curb scenario after being assessed by the proposed method** 179

6.5 **Experiment Setup** 180

6.6 **Distance Error Distribution** 183

6.7 **Jen 2 test case as seen by the image input device sensor** 184

6.8 **Jen 2 post-assessment as measured by the proposed algorithm** 184

6.9 **Jen 2 ground truth** 185

6.10 **GRIMOND 2 test case as seen by the imaging sensor** 185

6.11 **GRIMOND2 test case as observed by the algorithm.** 185

6.12 **GRIMOND2 - Distance Centre** 186

6.13 **GRIMOND2 Min Safe Distance** 186

6.14 **GRIMOND2 Traversable Area.** 187

6.15 **KENEDYST1 test case as seen by the imaging sensor** 188

6.16 **KENEDYST1 test case as observed by the algorithm** 188

6.17 **KENEDYST1 - Distance Centre** 189

6.19 **KENEDYST1 Traversable Area** 190

6.20 **PEARS2 test case as seen by the imaging sensor** 191

6.21 **PEARS2 test case as observed by the algorithm** 191

6.22 **PEARS2 Traversable Area** 192

List of Tables

2.1 **Suone et. al. Results** 50

2.2 **Vision-based Systems Comparison Table** 65

2.3 **Non-vision-based Systems Comparison Table** 67

2.4 **Incline detection and assessment techniques comparison table** 79

2.5 **Stairs and curbs detection and localisation techniques comparison table.** 89

4.1 **Performance metrics for all the custom-trained network** 126

4.2 **Detection accuracy comparison table for all three networks** 127

4.3 **Experiment Results.** 133

Nomenclature

Acronyms / Abbreviations

ABSAR	AI-Based Socially Assistive Robotics
ADA	Americans with Disabilities Act
AI	Artificial Intelligence
ALG	Algorithm
AT	Assistive Technology
ANFIS	Adaptive Neuro-fuzzy Inference System
AP	Average Precision
AUC	Area Under the Curve
BL	Bottom Left
BR	Bottom Right
C	Centre
CCD	Charge-Coupled Device
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DL	Deep Learning
DNN	Deep Neural Network
EPW	Electric-Powered Wheelchair
FHA	Federal Highway Administration
FN	False Negative

FP	False Positive
FPM	Frames Per Meter
FPR	Frame Processing Rate
FPS	Frames Per Second
GMA	Gaussian Mixture Algorithm
GPU	Graphical Processor Unit
GT	Grount Truth
Gyro	Gyroscope
HOG	Histogram of Oriented Gradients
IMU	Inertial Measurement Unit
IRI	International Roughness Index
IoU	Intersection over Union
KCC	Kent Country Council
LED	Light Emitting Diode
LiDAR	Light Detection And Ranging
LTPP	Long-Term Pavement Performance
mAP	mean Average Precision
MLESAC	Maximum Likelihood Estimation Sample Consensus
ML	Machine Learning
MPH	Miles Per Hour
MSX	Multispectral Dynamic Imaging
NED	Not Enough Data
NM	Not Mentioned

NN	Neural Network
NR	Not Relevant
OD	Object Detection
P	Precision
PMV	Personal Mobility Vehicles
PGM	Probabilistic Generative Model
R	Recall
RANSAC	Random Sample Consensus
R-CNN	Region-Based Convolutional Neural Network
RGB	Red Green Blue
RGB-D	Red Green Blue – Depth
ReLU	Rectified Linear Unit
RoI	Region of Interest
SLAM	Simultaneous Localisation And Mapping
SVM	Support Vector Machine
TL	Top Left
TR	Top Right
TN	True Negative
ToF	Time of Flight
TP	True Positive
UKC	University of Kent in Canterbury
YOLO	You Only Look Once
WHO	World Health Organisation

Chapter 1

Introduction

The unrelenting march of technological progress, particularly the phenomenal advancements in Artificial Intelligence (AI), has ushered in an era of seemingly limitless capabilities. Regardless of functionality, AI empowers universal systems to tackle an ever-expanding range of tasks, irrespective of their complexity. This transformative power extends to the realm of autonomous platforms, including self-driving vehicles and robotics-based modes of transportation. These advancements hold immense promise for everyday use, fostering a more seamless integration into our everyday lives.

One of AI's most profound impacts is being felt in the field of assistive technologies (AT). A term that encompasses a broad spectrum of hardware and software systems designed to empower individuals with disabilities. These technologies aim to bridge the gap in functional capabilities, fostering greater inclusion within society. By promoting independence and participation in daily life activities, AT plays a pivotal role in enhancing the overall well-being of people with disabilities. One of the most common limitations affecting individuals with disabilities is mobility. Persons with mobility-related disabilities often rely on mobility-related ATs such as mobility scooters and Electric-Powered Wheelchairs (EPWs). This thesis is focused on mobility-related ATs with the aim of embracing AI-related technologies and harnessing their power in order to improve the

quality of life for individuals with mobility impairments by facilitating greater ease and efficiency in their daily commutes and overall mobility. The proposed systems are versatile in their nature and can be deployed on numerous different robotic platforms, including but not limited to mobility-related assistive technologies.

This thesis is an integral part of the ABSAR (AI-Based Socially Assistive Robotics) [1], a project that investigates the application of AI techniques for robotic systems to govern electric-powered wheelchairs. This project endeavours to develop adaptive intelligent systems designed for seamless integration into Electric-Powered Wheelchairs (EPWs). These AI-powered enhancements aim to empower individuals with disabilities by fostering greater independence and facilitating a more enriching daily life. The integrated systems will provide autonomous navigation capabilities, allowing EPWs to traverse environments with minimal manual control. Additionally, environmental understanding features will equip EPWs to perceive and respond to their surroundings, ensuring a safer, smoother, and seamless user experience.

This thesis focuses on investigating the application of deep learning (DL) techniques fused with machine vision to address core challenges that mobility-related assistive technology users are constantly exposed to. Those challenges are negative obstacles, a term that refers to objects or features that lie below the ground plane, contrasting with positive obstacles that are situated above the ground. potholes, curbs, downward-facing stairs, and inclined platforms such as wheelchair ramps, dropped curbs, and other inclined planes that can be traversed by a wheelchair or a mobility scooter.

This goal is achievable via the reliance on real-time visual feedback provided by the system and returning accurate information relating to the negative obstacle in question. This feedback also includes an assessment of the traversability in order to assist the user in deciding whether it is safe to traverse the road before them or not. On the other hand, the system also provides sufficient accurate information to the user, informing them of the possible safe distance they can traverse before initiating the needed obstacle avoidance manoeuvres or, in certain cases, performing a complete early safe stop.

Just like other research journeys, numerous obstacles have been envisaged throughout the path towards the project's completion. These challenges range from data availability, such as the lack of proper datasets fit-for-purpose when the goal

is to detect negative obstacles, such as potholes in real-time, to having a reliable technique to detect safe traversable paths, to safely, accurately assess upwards and downwards-facing inclinations, and to detect and avoid the end of paths and dangerous drops such as curbs and downward-facing stairs. The solutions proposed in this thesis are aimed at overcoming the described challenges via the use of a lightweight, accurate, and robust deep learning object detection and localisation algorithm fused together with depth imaging and environment assessment and recognition in order to create a reliable and versatile system that can be mounted onto any moving platform capable of performing as infallibly as possible in indoor and outdoor scenarios regardless of the time, weather, location, or other possible limiting factors.

This thesis delves into the prevalent challenges, proposing innovative solutions to address them. The introduction chapter lays the groundwork for this exploration. It commences with a compelling discussion of the research motivations within section 1.1, highlighting the impetus for this work. Aims and objectives are then precisely articulated in section 1.2, followed by the core research questions the thesis endeavours to answer, which are laid out in section 1.3. The thesis emphasises its novel contributions and potential to advance the field in section 1.4 and envisions the practical applications and benefits of the proposed systems in section 1.5. The scope of the research is delineated in section 1.6, ultimately leading to a concise overview of the thesis structure within section 1.7, providing an as clear as possible roadmap for the reader.

1.1 Research Motivation

The field of Artificial Intelligence (AI) is experiencing a period of explosive growth fuelled by breakthroughs in machine vision. Innovations that once resided in the realm of science fiction, such as deep learning, object recognition, and environmental understanding, are now the cornerstones of groundbreaking real-world applications. This

progress extends beyond the realm of software. The evolution of powerful yet energy-efficient processors and Graphics Processing Units (GPUs) has paved the way for the seamless integration of sophisticated AI systems into compact robotic platforms. These groundbreaking technological strides hold immense potential to revolutionise the lives of individuals facing challenges that hinder their full participation in society.

By harnessing the power of AI and machine vision, assistive technologies such as electric-powered wheelchairs and mobility scooters can be transformed. Wheelchairs equipped with cutting-edge features like obstacle avoidance, allowing users to navigate complex environments with greater confidence and independence, are realistically possible with the help of such technologies. Autonomous navigation capabilities could further empower individuals by enabling them to travel predetermined routes or explore unfamiliar surroundings with minimal assistance. Advanced safety systems could provide an extra layer of protection, safeguarding users from potential hazards. These advancements, far from science fiction, have the potential to significantly mitigate the impact of mobility-related disabilities, fostering a more inclusive and accessible world for all.

This fuels a compelling motivation to improve the status quo by designing and implementing the systems discussed in this thesis. To design and implement the advanced systems explored in this thesis. Equipping users with such intelligent safety features has the potential to significantly reduce the barriers hindering their mobility. These systems could mitigate the risks associated with falls, injuries, and even fatalities caused by encounters with unexpected negative obstacles or downward inclines or stairs. This envisioned impact also draws inspiration from the growing accessibility and availability of driverless cars in the market. The prospect of similarly autonomous or semi-autonomous assistive technologies for individuals with disabilities serves as a powerful motivator for the implementation of the systems proposed in this thesis. Furthermore, their introduction could act as a springboard for further advancements and breakthroughs in the field of autonomous mobility assistance.

1.2 Research Questions

This thesis seeks to answer the following core research questions:

- What impact does the variety of object representation within a negative road anomalies dataset have on the performance of the object detection network? Given that object detection systems are widely available using different technologies and setups, how can the optimal technique be selected?
- How can deep learning-based object detection networks be utilised to accurately detect and classify potholes of varying sizes, shapes, and textures under diverse environmental conditions (e.g., lighting, weather, road surface) and in real-time?
- How can the geometric properties extracted from point cloud data be harnessed to detect and assess the accessibility of positive and negative inclined planes, including wheelchair ramps, in real-time scenarios?
- How can a point cloud segmentation algorithm be designed to efficiently and accurately isolate curb and downward-facing stair edges in real-time scenarios within the real world?
- How can the reliability and robustness of the proposed systems be rigorously evaluated under diverse and unpredictable real-world conditions, taking into consideration indoor and outdoor scenarios in certain cases?
- How can the proposed systems be utilised for the purpose of providing an additional layer of safety to traditional mobility-focused assistive technologies? What benefit do they introduce?

1.3 Aim and Objectives

This thesis investigates the potential of leveraging deep learning-based object detection techniques along with computer vision systems for comprehensive, real-time analysis of diverse negative obstacles with respect to mobile platforms, regardless of their nature.

The key objectives of this thesis are set out below:

- To investigate the viability of developing and implementing intelligent computer vision systems powered by deep learning enhanced with a spatial understanding of the environment capable of detecting, and localising negative road anomalies in real-time.
- To investigate the viability of developing and implementing a computer vision-based system capable of assessing the traversability of different paths, taking negative inclinations, curbs, and downward-facing stairs into consideration.
- To assess and examine the dependability, accuracy, and resilience of the introduced systems by employing rigorous testing scenarios.
- To deploy and integrate the proposed systems within real-world applications and scenarios. Electric-powered wheelchairs represent a highly suitable mobile platform for the deployment and testing of such systems.

1.4 Applications

The central goal of this thesis is the development of a versatile framework designed for seamless integration with a range of mobility-related assistive technologies (ATs). These ATs primarily encompass electric-powered wheelchairs and mobility scooters, two crucial means of mobility for individuals with mobility impairments to navigate their surroundings independently. The proposed system aims to significantly enhance the situational awareness of these ATs by leveraging sensor fusion, a technique that combines data from two or more imaging modalities. This multi-sensor approach can potentially address limitations inherent in any single sensor, providing a more comprehensive and robust understanding of the environment.

The system's output offers a unique versatility, catering to two distinct yet complementary functionalities. It can firstly provide real-time guidance for users who operate the ATs manually or in a semi-autonomous mode. This can involve visual or haptic feedback mechanisms that alert users to potential obstacles or path deviations. Secondly, the system's processed data can serve as a crucial input for a fully autonomous navigation algorithm. This paves the way for the development of intelligent ATs that can navigate environments independently, significantly improving user mobility and autonomy.

Following initial deployment and rigorous testing within controlled indoor and outdoor environments, the system's adaptability has been demonstrably validated. This adaptability allows for customisation to cater to diverse scenarios and the detection of a wide range of target objects. Notably, the system can be configured to prioritise specific objects relevant to the user's needs, such as identifying accessible doorways or avoiding obstacles like curbs or uneven terrain. To ensure generalisability, the system has been subjected to thorough evaluation under a variety of real-world conditions. These evaluations have consistently yielded positive results, confirming the system's high

performance, reliability, and remarkably low error rate. This combination of adaptability, versatility, and robust performance positions the proposed framework as a significant advancement in the field of assistive technologies.

1.5 Main Contributions

- Designed, implemented, and tested a system that relies on deep learning object recognition and depth data in order to detect negative obstacles such as potholes in real-time and in real-life scenarios. The system includes a novel methodology that localises the detected objects and enables them to be treated as positive obstacles, facilitating their integration into obstacle avoidance algorithms.
- Designed, implemented, and tested a system that uses depth data, mainly point clouds, to segment and assess positive and negative inclined planes, such as wheelchair ramps, in real time to assess their traversability.
- Designed, implemented, and tested a system that uses depth data to segment the ground plane, detect its edge, and avoid it in different scenarios such as curbs, downward-facing stairs, etc.
- Manually connected and annotated a widely covering dataset of pothole images that can be used for training and evaluating object detection algorithms. The dataset is made publicly available for future reference.
- The proposed systems offer reconfiguration and retraining capabilities, enabling the detection of diverse negative obstacles regardless of their form, placement, surface properties, composition, or environmental conditions. This presents a novel framework for negative obstacle detection and localisation with potential for wide-ranging applications.

1.6 Scope of the Thesis

This thesis delves into the development of a novel computer vision framework designed to revolutionise navigation and scene understanding for individuals with disabilities. This novel system transcends the limitations of traditional navigation systems by leveraging cutting-edge computer vision techniques. The proposed system can be seamlessly integrated into existing mobile platforms, such as powered wheelchairs or mobility scooters. By equipping mobility-related assistive technologies with this smart system, users would gain the ability to comprehend their surroundings in real-time. The framework would analyse visual data from the environment, detecting and classifying negative objects, obstacles, and pathways. This enhanced awareness would empower users to navigate diverse indoor and outdoor environments with greater confidence and independence, whether manoeuvring through crowded corridors or navigating uneven outdoor terrains. The developed systems must exhibit adaptability and customisation capabilities to achieve this transformative objective. The framework should be able to learn and adjust to individual user preferences, such as preferred cruising speeds or safety tolerances. Additionally, transparency and reliability are paramount attributes.

The scope of this thesis is set out below:

- This thesis leverages a pre-existing deep learning (DL) neural network for object detection and recognition. Consequently, the design and implementation of a new object detection network therefore falls outside the scope of this work.
- This thesis does not directly address autonomous systems collision avoidance and path tracking, as these areas lie outside its scope.
- Deep Learning (DL) explanation and visualisation techniques, along with model approximation methods, are beyond the scope of this thesis as they typically focus on

improving model efficiency or interpretability. The high accuracy achieved, as evidenced by the results, suggests that further optimisation of the DL object detection technique may not be necessary for the current investigation.

- This research focuses on evaluating the traversability of various ground and inclined planes within the environment. Path planning algorithms are beyond the scope of this thesis.
- The detailed impact of vibrations on the proposed systems will be addressed in future work; it is outside of the scope of this thesis.

1.7 Structure of the Thesis

The structure of this thesis is set out as follows:

- **Chapter 2** presents an in-depth review of the negative obstacle detection systems classified within different groups. The first group being the state-of-the-art pothole detection systems organised as vision-based systems (deep learning systems and non-deep learning-based systems), and non-vision-based systems, the second being state-of-the-art inclined planes detection, segmentation, and assessment systems, and the third being curb, downward-facing stairs, and edge of the paths detection and localisation systems.
- **Chapter 3** presents a meticulously curated pothole dataset. The dataset was manually collected and annotated, ensuring high-quality ground truth labels. It encompassed a broad range of pothole characteristics (shape, depth, colour, filling, type) and was imaged under diverse lighting and weather conditions. This dataset will prove invaluable for benchmarking and advancing algorithms in object recognition, localisation, and semantic segmentation, with particular relevance to road infrastructure assessment and autonomous systems. It can be used for assistive technologies-related projects, robotics, or even road surveying.

- **Chapter 4** presents a novel algorithm for real-time detection and localisation of negative road anomalies, specifically potholes. The algorithm leverages a deep learning convolutional neural network (CNN) pre-trained on a meticulously curated pothole dataset (introduced in Chapter 3). This pre-training step ensures high confidence and low latency detection. Uniquely, the algorithm fuses pothole detection with a spatial understanding of the surrounding environment obtained via depth imaging.

  The chapter commences with a detailed benchmarking process. Three state-of-the-art object detection CNNs are evaluated on the pothole dataset, and the optimal network is selected based on performance metrics. This ensures that the chosen CNN is the most effective for the proposed implementation. Subsequently, the chapter delves into the training and validation process, a crucial step guaranteeing the accuracy and reliability of the CNN's results. The chapter then details the innovative localisation process achieved through depth imaging. This integration of depth data allows for precise spatial understanding of the pothole's position within the environment. Finally, the chapter presents a rigorous testing and evaluation process where the system's accuracy and performance are assessed to illustrate its strengths and weaknesses.

  The proposed algorithm offers significant advancements for several reasons. First, it introduces a novel localisation technique for negative obstacles, employing unconventional measurements. This enables any navigation system, whether autonomous, semi-autonomous, or manual, to treat these negative obstacles as positive ones. Consequently, the avoidance process becomes significantly less complex and more efficient. Second, the algorithm establishes a versatile framework. The CNN can be efficiently retrained to detect various negative road obstacles, transforming them into "positive" obstacles within the algorithm or from the user's perspective. This adaptability broadens the algorithm's applicability to other negative road anomalies.

- **Chapter 5** introduces an innovative algorithm designed for the detection, segmentation, and localisation of upwards and downwards-inclined planes. This detailed chapter outlines the implementation process and presents a rigorous

experimental methodology. The system was subjected to both real-world and controlled in-lab environments, enabling a comprehensive assessment of its performance in diverse conditions. Such a system holds significant value for various applications. In assistive technologies, robotics, and surveying, accurate detection and assessment of inclined planes are crucial for determining the safe traversal of mobile platforms. The proposed algorithm demonstrates exceptional accuracy, coupled with low latency and computational requirements. This optimised design allows for its integration into a wide range of systems with varying computational capabilities.

- **Chapter 6** presents a novel algorithm for real-time detection and localisation of the end of the path within a road. The system goes beyond simple edge detection, encompassing features like curbs and downward-facing stairs. It critically assesses the safe, traversable distance between the mounted platform and the detected edge. The chapter delves into the technical details of the algorithm, outlining the various real-time measurements employed to ensure accuracy and safety throughout the navigation process. A rigorous assessment process is presented to comprehensively evaluate the system's performance. Testing encompasses diverse locations, time periods, and weather conditions. This multifaceted approach ensures the system's reliability and generalisability across various real-world scenarios.

  The proposed algorithm offers significant value for a broad spectrum of platforms. Its applicability extends from assistive technologies such as electric and manual wheelchairs and mobility scooters to larger platforms like cars and other transportation systems. By enabling robust road edge detection and localisation, the algorithm ensures these platforms maintain a safe braking distance from the road/path edge, thereby enhancing overall safety in navigation tasks.

- **Chapter 8** serves as the culminating point of the thesis. It revisits the central research questions and demonstrates how the findings offer compelling answers or solutions. The chapter also highlights the broader contributions of the work to the field and identifies promising areas for future research endeavours

Chapter 2

A Review on Negative Obstacles and Inclines Detection

2.1 Introduction

Navigation within complex environments is a fundamental challenge for nearly all types of moving platforms, whether on-road vehicles, assistive technologies (AT) [2], autonomous or semi-autonomous systems such as robots, self-driving vehicles [3], or assisted mobility devices [4]. These different systems must operate in a wide range of environments, from structured indoor settings in the case of assistive technologies or robotic platforms to unpredictable outdoor landscapes with respect to robotic platforms and self-driven and semi-autonomous vehicles.

Assistive technologies (AT) designed to enhance mobility encompass a range of devices. These include traditional wheelchairs (manually propelled devices), electric-powered wheelchairs (offering motorised movement), and mobility scooters (generally three or four-wheeled, steered with a tiller). The World Health Organisation (WHO), in its 2023 report, estimates that at least 80 million people globally, representing approximately 1% of the population, utilise wheelchairs for mobility assistance [5]. However, significant barriers to unobstructed access persist, including the lack of infrastructure adapted for

wheelchair use, uneven surfaces, potholes and other types of road anomalies. Current innovations in AT focus on advancements in powered mobility options and the integration of smart technology for increased user independence.

Current perception systems employed in autonomous technologies, such as self-driving vehicles and assistive robots, often prioritise the detection and avoidance of "positive obstacles" – physical objects protruding into their environment. However, this focus neglects the equally critical challenge posed by "negative obstacles" (e.g., potholes, downwards-facing stairs, curbs) [6] [7] and inclines (e.g., wheelchair ramps and inclined road surfaces) is critical for ensuring safe and efficient traversal of varied terrain. Unlike positive obstacles, which typically result in collisions, negative obstacles can cause falls, system damage, or mission failure. The consequences of encountering negative obstacles can be particularly severe for ATs designed to assist individuals with mobility limitations. For instance, a wheelchair traversing a curb without proper safety features or encountering a deep pothole is at risk of tipping over, potentially causing user injury and equipment damage. Similarly, self-driving vehicles navigating an undetected pothole could sustain suspension damage or veer off course, jeopardising passenger safety. dangerous as accidents caused by a collision with a positive obstacle are the ones resulting from a fall off of a curb or on downwards-facing stairs or a ramp without safety borders are a completely higher level of risk. Not to forget the dangerous risks caused by driving through a deep pothole. The wheelchair might fall, and the wheels could be broken, putting the user at a severe risk of harm and injury.

This highlights a critical limitation in current state-of-the-art autonomous, semi-autonomous, and guidance-assisted systems. To ensure the safe and reliable operation of ATs, robust algorithms capable of accurately identifying and navigating negative obstacles and inclines are crucial. Research efforts should prioritise the development of such capabilities to enhance the safety and efficacy of autonomous technologies across diverse operational environments.

This chapter focuses on the challenges and current novel advancements in the field. It will delve into the important and critical motivation behind the use of assistive technologies (2.2), followed by a detailed examination of the limitations of existing negative road anomalies detection and localisation technologies (2.4), incline estimation

and assessment techniques (2.4), the ground plane, curbs, and downwards-facing stairs detection and localisation techniques (2.5). This chapter then concludes by exploring future research directions and outlining potential systems that hold promise for achieving the stated research goals.

2.2 Assistive Technologies, the motivation

Assistive technologies, such as electric-powered wheelchairs (EPWs), offer multifaceted benefits to individuals with disabilities. Beyond the central advantage of improved mobility, EPWs facilitate increased productivity, participation in leisure activities, and enhanced independence [8]. Therefore, the design of assistive devices should focus on maximising user autonomy and enriching quality of life. Inadequately designed or malfunctioning assistive technologies, however, can significantly detract from the user experience [8].

Clinical observations indicate a near parity between patients able and unable to operate electric-powered wheelchairs (EPWs) [9] independently. Navigation within confined indoor spaces, particularly when manoeuvring through doorways, presents significant challenges for users. Furthermore, clinicians report that 40% of EPW users experience difficulty with steering tasks, with 5-9% finding them entirely unmanageable. A substantial 85% of individuals unable to operate EPWs independently are attributed to factors such as visual impairments, cognitive limitations, or motor skill deficits. The potential of automated navigation systems to reduce this percentage by half offers promising implications for greater accessibility and user autonomy [9].

Computer vision-based navigation systems present a transformative opportunity for enhancing the autonomy and safety of electric-powered wheelchair (EPW) users, offering both semi-autonomous and fully autonomous functionality. Innovative approaches include control mechanisms that leverage facial tracking [10] as well as precise eye and iris movement detection [11, 12]. Additionally, collision detection and avoidance technologies play a crucial role in assisting EPW users with obstacle negotiation [13, 14].

A navigation system utilising 3D stereo-vision to detect potential object collisions proactively was notably proposed [15], halting movement to prevent impact. Their system further incorporates visual odometry for path planning towards user-defined goals. Importantly, it offers adaptive navigation prompts tailored to the user's level of awareness, ensuring optimal support. A thorough examination of the most recent research directions in smart wheelchairs concluded that smart wheelchairs introduce a promising improvement to the well-being of their users as long as the users are able to confidently trust the system [16]. Another recent comprehensive review was conducted with the aim of studying the design and implementation of the sensory systems that can be used within EPWs, highlighting the importance of sensor fusion, novelty, interaction between the user and the system, cost, and other factors. These are to be considered critical for the advancement of smart assistive technologies [17].

The evolution of the state-of-the-art systems, as expected, did not stop at positive obstacles, as it was realised that although safe autonomous or semi-autonomous navigation of EPWs requires positive obstacle avoidance, the objective cannot be achieved without taking negative obstacles into consideration. Hence, negative obstacles were also addressed within the technological advancements. Some attempts to tackle the problem were made via the use of thermal imaging to detect negative obstacles [18], other attempts were made using the cloning-based approach to traverse negative obstacles [19]. As innovative as these systems are, they fall behind expectations when the platform to be utilised as a host to the system is an EPW or a mobility scooters. The mentioned system in [18] would fail in the case of water-filled negative obstacles, or during significantly hot days, whilst the system proposed in [19] would not be usable in the case of critical health-related systems such as assistive technologies as they would require obstacle avoidance rather than safe traversal due to the significant risk of harm to which users could be exposed.

This showcases the importance of employing advanced sensory and navigation systems on assistive technologies in order to ensure safe and reliable navigation. It also highlights the importance of addressing the negative obstacles problem, as a system cannot be considered safe and cleared for a fully autonomous, semi-autonomous, or assisted navigation if it does not address the negative obstacles problem as these obstacles,

in their nature, are considered a significantly challenging task due to the fact that they cannot be detected with traditional positive obstacle detection methods.

The proposed system, unlike others, aims at harnessing the benefits of sensor fusion, with deep learning, computer vision, and spatial understanding of the environment in order to detect, localise, and assess negative obstacles. The system is capable of accurately detecting and localising negative road anomalies in real-time, it is also capable of detecting and assessing upwards and downwards inclined planes to decide whether they are safe to traverse or not, it is also capable of accurately assessing the traversable area within a normal flat ground taking into consideration the end of the traversable area where a negative obstacle would be present either in the form of a curb, or a set of downwards-facing stairs.

Subsequent sections delve into contemporary systems engineered to mitigate the risks posed by negative road anomalies, with a particular emphasis on potholes. The detection and skilful circumnavigation of such obstacles are underscored as paramount for safe navigation. Moreover, a meticulous examination is conducted on systems capable of evaluating the traversability of both upward and downward inclined planes. This is followed by, a discussion illuminating novel methodologies within path traversability analysis, encompassing the complexities of curbs and downwards-facing stairs.

2.3 Negative Road Anomalies Detection Techniques

Negative road anomalies, such as cracks and potholes, present a significant obstacle to autonomous, semi-autonomous, and manual navigation of systems due to their inherent variability and the limitations of current sensor technologies and detection algorithms. The stochastic nature of these anomalies – their unpredictable shapes, depths, locations, and environmental contexts (e.g., lighting, weather, standing water) – renders traditional sensing techniques prone to false negatives. This challenge is compounded by the fact that anomalies occur year-round in public spaces, where detectors may be adversely affected by factors like light intensity, fog, or rain. Furthermore, their random patterns and depths pose difficulties for techniques

optimised for specific forms. The potential for standing water within anomalies further obfuscates detection. Timely identification of negative road anomalies is essential for efficient road maintenance, enhanced safety and comfort in autonomous driving, the well-being of wheelchair users, and the general expansion of safe vehicle automation.

This section presents a critical analysis of contemporary techniques, offering a thorough and objective assessment of their performance, reliability, and scalability. To facilitate a methodical comparison, the strengths and weaknesses of each technique will be systematically evaluated. Additional details about every assessed system can be found within the Appendices respectively.

2.3.1 Review Strategy and Performance Measures

In this section, a critical review of existing detection and localisation techniques for negative road anomalies detection and localisation will be conducted. Each method will be summarised with a focus on its underlying technology. Strengths and weaknesses will be evaluated based on key design criteria, including accuracy, precision, environmental robustness, and computational efficiency. The goal is to synthesise insights from this analysis, allowing the development of a novel detection and localisation algorithm that leverages the strengths of current techniques while offering superior accuracy and reliability. Where available, performance metrics such as Accuracy, Precision, Recall, and F1-Score will be extracted from the research literature to support the evaluation.

1. Accuracy: Ratio of the correctly predicted observations to the total observations:

$$\text{Accuracy}=\frac{TP+TN}{TP+FP+FN+TN} \tag{2.1}$$

2. Precision: Ratio of the observations correctly predicted divided by the total positive observations predicted:

$$\text{Precision (P)}=\frac{TP}{TP+FP} \tag{2.2}$$

3. Recall: Or sensitivity, Ratio of the positive observations correctly predicted divided by the total observations:

$$\text{Recall (R)} = \frac{TP}{TP+F} \qquad (2.3)$$

4. F1-Score: Average of the Precision and Recall:

$$\text{F1-Score} = \frac{2 \times (R \times P)}{(R + P)} \qquad (2.4)$$

In addition to performance measures, the following technical criteria will be evaluated: efficiency, real-time functionality, computational requirements, power consumption requirements, system size, and suitability for mass production.

2.3.2 Vision-Based Systems

2.3.2.1 Deep Learning-Based Techniques

Deep learning, as described by F. Chollet [20], is a foundational subfield of machine learning and artificial intelligence. It centres on the use of artificial neural networks composed of successive "layers" that extract increasingly abstract and meaningful representations from raw data. These networks, while loosely inspired by the biological structure of the brain, function as complex mathematical models. Each layer within a deep learning architecture acts as a filter, progressively refining the input data to isolate the features most relevant to the desired outcome. This hierarchical, multistage process allows deep learning models to "learn" complex patterns and relationships, often surpassing the accuracy of traditional machine learning technique.

The recent surge in Artificial Intelligence (AI) research has been fueled by several factors, including: (1) increasingly successful applications demonstrating AI's potential, (2) significant advancements in processing power enabling more complex models, (3) the growing availability and accessibility of vast datasets, and (4) the positive impact of AI

research on various scientific fields. This confluence of advancements has spurred significant interest in utilising machine learning (ML) techniques for automated detection of negative road anomalies, particularly potholes. This focus stems from the substantial economic and safety concerns associated with road surface deterioration.

Visible-Light RGB Camera as Input:

Pereira et al. [21] proposed a deep learning solution for pothole detection, specifically designed to be cost-effective. Their approach utilises a convolutional neural network (CNN) architecture to extract and analyse image features relevant to pothole identification. The proposed network consists of four convolutional and pooling layer pairs, followed by a fully connected layer. A Rectified Linear Unit (ReLU) serves as the activation function within this layer of the form $f(x) = \max\ (x\ , 0)$, known for its computational efficiency and ability to mitigate the vanishing gradient problem. The output layer employs a sigmoid function $S(x) = \frac{1}{1+e^{-x}} = \frac{1}{e^{x}+1}$, introducing non-linearity and mapping outputs to the probability of pothole presence. The authors likely experimented with various hyperparameters, including filter types, to optimise the network's performance.

The obtained result can be summarised as follows:

- Accuracy: 99.8%
- Precision: 100%
- Recall: 99.6%
- F1-score: 99.6%

As per the assessment criteria described earlier, this method is evaluated as follows:

- **Efficiency:** Results indicate strong performance efficiency.
- **Real-time Functionality:** Currently limited to still images, hindering real-time use.
- **Computing Power:** Manageable with specialised processors (e.g., Intel Neural Compute Stick). However, potential issues arise with larger image sizes.
- **Power Consumption:** Varies based on processor and overall equipment choice.
- **System Size:** Adaptable due to the nature of deep learning systems.
- **Mass Production:** Feasible, with the high-resolution camera as a potential cost bottleneck.

Anand et al. [22] propose a deep learning-based method for detecting cracks and potholes in road surfaces. Their method leverages convolutional neural networks (CNNs) to analyse textural and spatial features. The system comprises several stages:

- **Road Segmentation:** A SegNet network [23, 24, 25] isolates the road surface from the image.
- **Edge Detection and Enhancement:** Canny edge detection generates a second mask, followed by edge dilation to enhance connectivity.
- **Candidate Region Extraction and Filtering:** The combined masks define candidate regions, resized to 64x64 patches. Filtering removes false positives (e.g., shadows, vehicles).
- **Feature Encoding:** The authors replace the final convolutional layer of SqueezeNet [26], a compact CNN architecture, modified version of [27], with a custom dictionary-based residual encoding layer. This layer facilitates weighted assignment of image descriptors to codewords, acting as a pooling mechanism.
- **Classification:** Supervised learning trains the network on labelled data. A fully connected layer performs classification, followed by a Softmax layer for mutually exclusive class assignment. Binary cross-entropy is used as the loss function.

Based on the established criteria, the system demonstrates the following characteristics:

- **Efficiency:** Accuracy ranges from 92.37% to 99.99%. While achieving high accuracy in certain contexts, this variability indicates potential limitations that may impact overall efficiency in real-world deployment.
- **Real-time Functionality:** The system is capable of processing both images and videos, making it suitable for real-time applications.
- **Computing Requirements:** Compatibility with portable deep learning processors (e.g., Intel Neural Compute Stick) suggests manageable computational demands.
- **Power Consumption:** Power needs are dependent on the specific processor and associated hardware. A thorough power analysis is required to determine suitability for the intended use case.
- **System Size:** The deep learning architecture allows for flexibility in system size based on hardware choices.
- **Mass Production Potential:** The general availability of required components supports potential mass production. However, the cost and performance implications of a high-resolution camera necessitate careful consideration.

Gopalakrishnan et al. [29] propose a method for automated pavement crack detection utilising transfer learning with pre-trained deep learning models. Their approach leverages image datasets from the Federal Highway Administration (FHA) and the Long-Term Pavement Performance (LTPP) program in the United States, along with a Canadian database. The dataset is comprised of 1056 images organised as follows:

- Training: 760 images
- Validation: 84 images
- Testing: 212 images

Gopalakrishnan et al. employ a transfer learning approach utilising the Keras implementation of VGG-16 [30], a deep convolutional neural network (CNN) pre-trained on ImageNet [31].

The obtained result via the use of a Single NN Classifier can be summarised as follows:

- Accuracy: 90.0%
- Precision: 90.0%
- Recall: 90.0%
- F1-score: 90.0%

Based on established criteria, the system demonstrates the following characteristics:

- **Efficiency:** With 90% reported accuracy, the system exhibits partial efficiency. However, this metric alone doesn't guarantee consistent performance under real-world conditions.
- **Real-time Functionality:** The potential for real-time processing (images and videos) is promising.
- **Computational Requirements:** The ability to run on portable deep learning processors indicates manageable computational demands.
- **Power Consumption:** Power needs will vary based on specific hardware components.
- **System Size:** The deep learning architecture allows for flexibility in system size, adaptable to available equipment.
- **Mass Production:** Feasibility for mass production is likely, given the availability of standard components. The potential cost implications of a high-resolution camera require further consideration.

Suong et al. [38] developed a pothole detection and identification system employing YOLO (You Only Look Once) version 2 as its deep convolutional neural network (CNN) foundation. Their work explored two YOLO architectural variants: the established Darknet YOLO v2 architecture [39] and a novel architecture proposed by the authors, designed to optimise computational efficiency and reduce model size. The proposed architecture achieves its goals with 18 million parameters, a significant reduction from the 48 million required by the Darknet counterpart

To facilitate multi-object detection, the authors integrated the Anchor Box Model, predicting $(5 + numberOfClass) \times numberOfAnchorboxes$. Each box is specialised for detecting objects of varying sizes and aspect ratios. K-means clustering was applied to the training dataset, yielding five distinct anchor boxes tailored to the specific characteristics of the data. These dataset-oriented anchor boxes further refine the system.

Table 2.1: **Suong et. al. Results**

Model	**Average Precision**	**Recall**	**Parameters**	**Frame per Second**
YoLo v2	60.14%	65.61%	48 million	23
Authors' models and anchors	67.74%	74.93%	18 million	32
Authors model and Den-anchor	83.43%	83.72%	18 million	21

Based on established criteria, the system demonstrates the following characteristics:

- **Efficiency:** The reported accuracy of 82.43% indicates limited efficiency, potentially hindering reliable real-world performance.
- **Real-time Functionality:** While theoretically capable of real-time operation, insufficient testing raises concerns about performance under dynamic conditions.
- **Computational Requirements:** The suitability for deployment on compact deep learning processors suggests manageable computational demands.
- **Power Consumption:** Power requirements are contingent upon specific hardware and peripherals; detailed analysis is warranted

- **System Size:** The deep learning architecture allows for flexibility in physical form factor, though hardware constraints must be considered.
- **Mass-Production Feasibility:** The system's reliance on standard components supports potential mass production. However, the cost and availability of high-resolution cameras pose a crucial economic consideration.

Furthermore, while recent studies utilising YOLO networks for pothole detection demonstrate promising results [40, 41, 42], relying on publicly available datasets such as [43, 44, 45] several limitations warrant consideration:

- **Precision Limitations:** Maximum-reported mean average precision (mAP) values of 94% suggest room for improvement in system reliability, as false positives or missed detections could still occur.
- **Localisation Scope:** The focus of these systems is primarily on detection; precise localisation functionalities remain to be addressed to provide actionable guidance for avoidance or repair.
- **Computational Demands:** Reliance on the full YOLO architecture often necessitates high computational power and may limit real-time frame rates, impacting practical deployment.
- **Robustness to Occlusions:** The ability to handle water-filled potholes, a common real-world challenge, is not addressed in these studies.
- **Dataset Inadequacy:** Insufficient testing data limits the ability to draw definitive conclusions about system performance in diverse real-world scenarios.

In summary, while these efforts represent progress in the field, advancements in precision, localisation, computational efficiency, robustness to occlusions, and more comprehensive real-world testing are needed for reliable practical applications.

Based on the discussed criteria, these attempts would be assessed as follows:

- **Efficiency:** The limited mean average precision (mAP) (<94%) indicates potential for false positives or missed detections, impacting overall efficiency.
- **Real-time Functionality:** Theoretical real-time potential exists, but rigorous testing under dynamic conditions is required for validation.
- **Computational Requirements:** The use of a full YOLO network suggests substantial computational demands, potentially necessitating specialised hardware.

- **Power Consumption:** Comprehensive power analysis is needed, as the choice of processor and peripherals will significantly impact consumption.
- **System Size**: While the deep learning architecture offers flexibility, hardware requirements and the necessary camera specifications must be considered for real-world deployment**.**
- **Mass-Production Feasibility:** The reliance on high-end components for real-time YOLO execution raises cost concerns for potential mass production.

Thermal Imaging as Input:

Aparna et al. [46] introduce a real-time pothole detection system leveraging thermal imaging and convolutional neural networks (CNNs). Their methodology comprises several key components:

1. Custom CNN Architecture:

- **Sequential Design:** The core of their approach is a sequential CNN model where normalised input undergoes processing by a series of 2D convolutional layers. Each convolution employs a 3x4 kernel with ReLU activation.
- **Pooling and Normalisation:** Max pooling layers follow each convolution, with outputs subsequently normalised through batch normalisation. Global average pooling then prepares the data for the final classification stage.
- **Classification and Optimisation:** A dense layer with sigmoid activation performs binary classification (pothole or no pothole). The model utilises cross-entropy (logarithmic) loss and the Adam optimiser for training.

2. Transfer Learning with ResNet:

Model Adaptation: To complement their custom CNN, the authors explore transfer learning with various ResNet architectures [47]. This leverages the pre-trained features of ResNet models.

Training Refinement: Techniques like cyclic and differential learning rates, along with the Fastai library (built on PyTorch), are employed to optimise training, aiming for improved accuracy and reduced overfitting.

3. Image Acquisition and Enhancement:

Thermal Imaging: A FLIR ONE thermal camera, featuring patented Multispectral Dynamic Imaging (MSX) technology [48], is used for image capture. MSX blends thermal and visible image data to enhance detail.

Data Collection: Images are collected in Chandigarh, India, under diverse conditions (time of day, lighting, temperature). The 500-image dataset includes potholes (dry, water-filled, wet) and potential confounders like shadows.

Metadata: Each image is annotated with metadata including temperature measurements, severity classification, water presence, shade, location, and a unique identifier.

4. Preprocessing:

Image Preparation: Images undergo cropping, resizing, and data augmentation techniques (e.g., zooming, rotation, mirroring, blurring, contrast enhancement, noise addition). This prepares them for input into both the custom CNN and ResNet models.

Overall, the work presents valuable insights, but further research is needed to address the identified limitations and ensure the system's reliability across a wider range of environmental scenarios.

Based on established criteria, the system demonstrates the following characteristics:

- **Efficiency:** The reported accuracy of over 95%, along with the use of thermal imaging for robust detection, suggests high efficiency.

- Real-time Functionality: The potential for real-time operation is promising.
- **Computational Requirements:** Significant computational power is likely required for model training. However, deployment can be flexible, accommodating various hardware configurations.
- **Power Consumption:** The system appears to have modest power requirements, utilising a controller (potentially low-power options like Intel Neural Compute Stick or Raspberry Pi) and a thermal camera.
- **System Size:** The deep learning architecture allows for adaptability in system size, contingent on the chosen hardware.
- **Mass-Production Feasibility:** The system's reliance on readily available components (controller, thermal camera) and lack of proprietary software enhances its suitability for mass production.

Laser Imaging as Input:

Yu et. al. [49] suggested a different approach that uses the imaging processing technique in order to extract laser-coloured regions within an image.

The image processing pipeline begins with noise reduction and laser region extraction. A multi-window median filter, employing four masks, mitigates noise within the image. This specialised filter likely uses different window sizes or orientations to tackle various noise patterns. Next, Otsu's method [50] dynamically calculates a threshold to convert the image into a binary representation, separating the laser from the background. This intelligent algorithm analyses the image's histogram (pixel intensity distribution) to determine an optimal threshold that maximises the distinction between the two classes. To further refine this representation, morphological closing is employed. Dilation (expanding white laser regions) followed by erosion (contracting them) fills gaps, connects nearby laser pixels, and smooths boundaries without significantly altering the target area's size. This process uses a line-shaped structuring element (20 pixels) to match the laser's projection. Finally, connected components analysis labels groups of connected foreground pixels. Components with pixel counts

below a predefined threshold are deemed as residual noise and discarded, ensuring a cleaner representation of the laser region.

The final decision is then made via a neural network configured as follows:

- Number of input nodes: 4
- Number of hidden nodes: 8
- Number of output nodes: 5

This neural network deduces the distress classification as per the author's predefined guidelines.

Based on the provided information, the system exhibits the following characteristics:

- **Efficiency:** Insufficient data exists to evaluate the system's computational efficiency and suitability for real-time navigation.
- **Real-time Functionality:** The presented information is inadequate to determine whether the system can operate under real-time constraints.
- **Computational Requirements:** The reliance on MATLAB suggests significant computational demands. Resource-intensive deep learning frameworks and image processing pipelines typically require substantial processing power and memory.
- **Power Consumption:** The use of laser imaging equipment and MATLAB incurs a high-power consumption overhead. This presents challenges for deployment on mobile platforms with limited power resources.
- **System Size:** While deep learning architectures offer some flexibility in physical form factor, the specific hardware requirements remain to be detailed.
- **Mass-Production Feasibility:** The use of MATLAB, with its associated licensing costs, poses a significant barrier to cost-effective mass production. Additionally, the cost and availability of specialised laser imaging equipment further limit scalability.

2.3.2.2 Non-Deep Learning-Based Techniques

Visible-Light RGB Camera as Input:

Azhar et al. [51] introduced a supervised learning approach for the detection and localisation of potholes within asphalt pavement images. Their method analyses image features to classify image regions as either "pothole" or "non-pothole," while providing the pothole's location within the image.

The technique employs Histogram of Oriented Gradients (HOG) feature extraction, a method emphasising object shape representation. This involves calculating the frequency of gradient orientations within localised image portions.

The system was tested with the help of a dataset of 120 images [53] where 50 images were used for training leaving the remaining 70 images to be used for validation.

The results obtained were as follows:

- **Accuracy:** 90.0%
- **Precision:** 86.5%
- **Recall:** 94.1%

As per the chosen criteria, the system can be assessed as follows:

- **Efficiency:** The system is fundamentally inefficient. A high false-negative rate, low accuracy, and the inability to process high-resolution images significantly compromise its utility in navigation tasks.
- **Real-time Functionality:** The restriction to static image analysis precludes real-time operation. This severely limits potential for integration into autonomous navigation systems.
- **Computational Demands:** The system's excessive computational overhead (0.673 seconds for a low-resolution 200 × 200 image) underscores its impracticality. Modern image sensors generate far larger image data, making real-time performance unattainable.
- **Power Consumption:** High computational requirements translate directly into substantial power consumption, rendering the system incompatible with battery-powered platforms commonly used in mobile applications.

- **System Size:** While the core components (processor, camera) suggest a manageable physical footprint, the need for significant processing power indirectly limits the system's form factor.
- **Mass-Production Feasibility:** The potential need for a high-resolution camera poses a significant economic barrier to mass production. This cost factor could severely limit the system's scalability and market viability.

Koch et al. [53] proposed a pothole detection system combining image segmentation, shape analysis, and texture comparison. Their system employs a fish-eye camera mounted on a moving vehicle to capture road images. The core methodology consists of three primary stages, Segmentation, Shape Analysis, and Texture Comparison detailed within Appendix A.1.2

The results obtained were as follows:

- **Accuracy:** 95.9%
- **Precision:** 81.6%
- **Recall:** 86.1%

It is worth mentioning Ryu et. al. [55] whom attempted to replicate Koch et. al's method with some minor changes including "Candidate Region Extraction" and "Decision". Although the technique they attempted to implement was promising, the results were not as expected. Their performance was significantly less than Koch. et al. with an accuracy of 73.5%. Their system is also not usable in real-time scenarios which prevents it from being a technique to consider.

Schiopu et al. [56] developed a video-based pothole detection and tracking system designed for use with standard cameras. Their approach capitalises on the observation that potholes tend to appear as high-intensity (bright) regions within images

The system was evaluated using a Samsung Galaxy S4 front camera, collecting 34 minutes of footage under clear, dry conditions. Their MATLAB implementation

successfully detected 55 potholes with six false positives and zero false negatives, demonstrating the promising potential for real-world applications.

The results were as follows:

- **Precision:** 90%
- **Recall:** 100%

Although the system provides promising results, several key questions remain regarding its real-world viability:

- **Efficiency:** While promising, the current precision level necessitates improvement to ensure reliable pothole detection, a crucial factor for time-sensitive navigation tasks.
- **Real-time Functionality:** The video-based approach suggests real-time potential. However, rigorous testing is needed to determine whether detection and tracking can occur within the time constraints of real-world navigation scenarios.
- **Computational Requirements:** The reported runtime is encouraging, but a detailed analysis of resource usage (CPU, memory) is essential to assess suitability for resource-constrained platforms.
- **Power Consumption:** The reliance on a camera and processor suggests the potential for low power operation, but this needs to be empirically verified.
- **System Size:** The core components offer flexibility for physical implementation.
- **Mass-Production Feasibility:** The system's reliance on MATLAB poses potential challenges for mass production. Porting the algorithm to a non-proprietary framework would be necessary for wider scalability and cost-effectiveness.

Dihao et. al. [57] developed an innovative crack detection method that leverages the power of probabilistic modeling and multi-scale analysis. Their system begins by analysing image data at the pixel level. Using a Probabilistic Generative Model (PGM), they calculate a probability map where each pixel's value indicates its likelihood of belonging to a crack based on its intensity. The authors demonstrated that the PGM outperforms the traditional Otsu thresholding method [50] for this task. To complement this intensity-based analysis, the system also examines the neighbourhood surrounding each pixel. Using a custom algorithm, it generates a second probability map, capturing the likelihood of a crack based on local patterns and textures. Fusing these two probability maps allows the system to achieve enhanced detection accuracy. Finally, to address the issue of border pixels (which might have lower probabilities), the authors employ a weighted dilation technique. This technique leverages the probability information to selectively expand crack regions, improving continuity without artificially increasing crack width.

The system's results were as follows:

- **Precision:** 90.7%
- **Recall:** 84.6%
- **F1-Score:** 87.0%

Assessing the system within the specified criteria yields the following:

- **Efficiency:** Insufficient efficiency significantly compromises the system's viability for time-sensitive navigational tasks.
- **Real-time Functionality:** Presumed high computational demands pose severe challenges for real-time operation. The lack of runtime data from the authors further obscures the system's performance characteristics.
- **Computational Requirements:** The system appears to necessitate significant computational resources. A rigorous analysis of resource usage (CPU, memory) is essential to confirm this and to assess the feasibility of optimisation.
- **Power Consumption:** High computational requirements strongly suggest substantial power consumption. Empirical verification remains necessary.
- **System Size:** The core components offer flexibility for physical implementation.
- **Mass-Production Feasibility:** While fundamentally feasible, the mass production of this system likely hinges upon reducing computational overhead. The reliance on a powerful processor could significantly impact production costs.

Stereo Vision as Input:

He Youquan et. al. [59] proposed a pothole detection system based on three-dimensional projection transformation to extract geometric features of the pothole. Their approach combines image preprocessing (binarisation, thinning), 3D reconstruction, and error analysis. System calibration, achieved with binocular stereovision, establishes a mapping between real-world coordinates and image coordinates. Noise reduction via neighbourhood averaging and Otsu thresholding [50] prepares the image for light band extraction. Geometric calculations transform image coordinates into real-world coordinates, yielding pothole measurements. The authors report a 2mm measurement discrepancy.

The present information raises significant concerns about the system's real-world suitability while offering some insights into its logistical feasibility:

- **Efficiency:** Insufficient data exists to determine the system's computational efficiency and its potential for real-time navigational tasks.
- **Real-time Functionality:** Limited functionality (detection without precise localisation) and the lack of runtime data severely hinder the assessment of real-time viability. Additionally, the requirement for close proximity to the pothole introduces operational constraints.
- **Computational Requirements:** Based on the system's components, computational demands appear manageable for potential deployment on mobile platforms. However, a detailed analysis is unavailable.
- **Power Consumption:** The system's design suggests the potential for integration with a mobile platform's power source. Quantitative power usage data is needed to confirm.
- **System Size:** The core components (LED light, CCD camera, processor) indicate manageable physical dimensions.
- **Mass-Production Feasibility:** The system's fundamental design suggests scalability. However, the potential need for a high-resolution CCD camera could significantly impact production costs.

Zhang et. al. [60] developed a real-time pothole detection system employing stereo vision for depth perception. Their approach begins with a custom disparity calculation algorithm designed to determine the pixel-level shift between two camera views. This disparity data is crucial for reconstructing 3D information about the road. Next, they employ a robust least-squares fitting method [61, 62, 63] to estimate the typical road surface, even in the presence of noise. To model the potentially curved road surface, they define a mathematical equation. Regions deviating significantly from this model (beyond a 0.04m threshold) are classified as potholes. Connected component labelling is then used to isolate and identify individual potholes. To achieve real-time performance, subsequent work [64] introduced optimisations including algorithm streamlining (removing a noise filtering step, RANSAC-based sampling), efficient coding practices (lookup tables, loop unrolling), and code parallelisation using OpenMP for multi-core processors.

The achieved optimised results are as follows:

- **Accuracy:** 98.0%
- **Recall:** 100%

As per the established criteria, this system can be assessed as follows:

- **Efficiency:** Insufficient data exists to determine the system's computational efficiency, a crucial factor for real-time navigational tasks.
- **Real-time Functionality:** The presented methods suggest the potential for real-time operation, especially as optimisations were introduced. However, a rigorous runtime analysis is lacking.
- **Computational Requirements:** The system's components indicate manageable computational demands, potentially suitable for mobile platform integration. Detailed resource usage analysis is needed for confirmation.
- **Power Consumption:** The system's design suggests compatibility with a mobile platform's power source, but quantitative power consumption data is required.
- **System Size:** The core components (stereo camera, processor) imply a manageable physical footprint.
- **Mass-Production Feasibility:** The fundamental system design appears scalable for mass production.

Li et. al [65] developed a stereo vision-based pothole detection method designed to extract 3D geometric features. Their approach involves an offline calibration step and an

online detection pipeline. Offline, they employ Zhang's calibration method with a checkerboard pattern (8 × 6, 24.5mm squares) to determine intrinsic and extrinsic camera parameters. During online operation, the system first acquires images from the calibrated stereo setup. A custom algorithm then calculates a disparity map based on geometric relationships. Triangulation is used to re-project the disparity map into 3D space, yielding coordinates for each image point. To model the expected road surface, a bi-square weighted robust least-squares method [61] is employed, minimising the influence of outliers. Regions deviating significantly from this surface are labelled as potholes using connected component labelling [66] . The authors tested their system with two USB cameras mounted on a cart and a Raspberry Pi 2 for processing. Using 640 × 480 images, pothole detection took approximately 4.94 seconds.

The authors did not provide any data which relates to the success and failure rates of the system which makes assessing the performance of the system hard.

In summary, as per our criteria, the system's assessment is as follows:

- **Efficiency:** Not enough evidence has been provided in order to assess this criterion.
- **Real-time Functionality:** This system can be used in real-time.
- **Computational Requirements:** The systems' computing power required is manageable on a moving platform.
- **Power Consumption:** This system can be mounted to a moving platform's battery.
- **System Size:** This system's size is acceptable as it relies on a stereo imaging camera along with the required processor.
- **Mass-Production Feasibility:** This system can be eligible for mass-production due to the nature of the equipment needed.

Depth Camera as Input:

Moazzam et. al. [67] proposed a pothole detection system utilising a Microsoft Kinect depth sensor [68, 69, 70]. Positioned 0.8-0.9 meters above ground, the sensor provides data processed within MATLAB. Potholes are identified as local minima within each image column. Depth (Z-axis) is directly measured in millimetres, while pixel-based X and Y

coordinates are converted to real-world units using calibration parameters. MATLAB functions calculate descriptive statistics for detected potholes (mean depth, maximum depth, standard deviation).

The system's assessment according to the discussed criteria is as follows:

- **Efficiency:** While providing extensive measurements, the 15% error rate necessitates improvement for reliable deployment. Optimisation strategies should be explored to enhance accuracy without sacrificing computational efficiency.
- **Real-time Functionality:** Insufficient data exists to determine the system's real-time suitability. Runtime analysis on representative hardware is essential.
- **Computational Requirements:** The overall system design suggests manageable computational demands for potential use on mobile platforms. However, the reliance on MATLAB introduces potential bottlenecks and licensing requirements.
- **Power Consumption:** The system appears compatible with a mobile platform's power source, but detailed analysis would be needed to confirm this assumption.
- **System Size:** The core components (depth camera, processor) indicate a manageable physical footprint.
- **Mass-Production Feasibility:** The system's fundamental design suggests scalability. However, the MATLAB dependency poses a significant barrier to cost-effective mass production. Porting the core algorithms to an open-source framework or a language without licensing constraints would be crucial.

2.3.3 Non-Vision Based Systems

Tire Pressure / Vibration as Input:

Researchers have explored various approaches for pothole detection using smartphone sensors and specialised hardware. Some methods [71, 72, 73, 74] focus on changes in accelerometer readings, as potholes induce distinctive vibrations, along with potential use of ultrasonic sensors and GPS. Variations exist, such as analysing tyre sounds, pressure fluctuations [75, 76, 77] and even deep learning techniques [78]. A common goal is the calculation of the International Roughness Index (IRI). However, these methods share a fundamental limitation: they inherently rely on the vehicle having already encountered the pothole. This reactive approach precludes their use in real-time obstacle avoidance for autonomous vehicles.

The techniques described so far have been summarised in the comparison table below:

Table 2.2: **Vision-based systems comparison table**

Authors	Method	Performance				Is Deep Learning?	Main Limitation	Efficiency	Real-time	Computational Power	Power Needed	Size	Mass-produ
		Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)								
Pereira et al.	Camera + CNN	98.8	100	99.6	99.6	Yes	Can only be used on images	Efficient	No	Low for low-res images, manageable	Manageable	Manageable	Yes
Anand et. al. method	Camera + CNN	99.893	NM	NM	72.14	Yes	Texture-based	Partly-efficient	Yes	Manageable	Manageable	Manageable	Yes
Gopalakrishnan et. al.	Camera + DCNN	90.0	90.0	90.0	90.0	Yes	Lack of proper testing	Partly-efficient	Yes	Manageable	Manageable	Manageable	Yes
Azhar et. al	Camera + Supervised Learning	90.0	86.5	94.1	NM	No	Large amount of calculations	Not Efficient	No	Large amount	Manageable	Manageable	Yes
Koch et. al	Camera + Segmentation	85.9	81.6	86.1	NM	No	Low performance	Not Efficient	No	Large amount	Large amount	Manageable	Yes
Ryu et. al	Camera + Segmentation (improved Koch. et. al)	73.5	80.0	72.3	NM	No	Low Performance	Not Efficient	No	Large amount	Large amount	Manageable	Yes
Schiopu et. al	Camera + Thresholding	NM	90.0	90.0	NM	No	Different speeds of the platform	Acceptable	Yes	Acceptable	Manageable	Manageable	Yes
Dihao et. al	Camera + PGM + SVM	NM	90.7	84.6	87.0	No	Large computational power	No	No	Large amount	Large amount	Manageable	Yes bu cost

He Youquan et. al	Stereo Vision	NM	NM	NM	NM	No	Intensity of light	NED	NED	Manageable	Manageable	Manageable	Yes
Zhang et. al. + Mikhailiuk et. al (Improved)	Stereo Vision	98.0	NM	100	NM	No	Light Intensity + Water-filled potholes	Yes	Yes	Manageable	Manageable	Manageable	Yes
Moazzam et. al	Depth Camera	NM	NM	NM	NM	No	Intensity of light	Partly-efficient	NED	Manageable	Manageable	Manageable	Yes
Suone et. al	Camera	NM	82.43	83.72	NM	Yes	Low Performance	Not Efficient	Yes (not tested)	Manageable	Manageable	Manageable	Yes
Li et. al	Depth Camera	NM	NM	NM	NM	Yes	Light Intensity + Water-filled potholes	NED	Yes	Manageable	Manageable	Manageable	Yes

NM: Not Mentioned

NR: Not Relevant

NED: Not Enough Data

Table 2.3: **Non-vision-based systems comparison table**

Authors	Method	Performance				Is Deep Learning?	Main Limitation	Efficiency	Real-time	Computational Power	Power Needed	Size	Mass-production
		Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)								
Yu et al.	Laser Imaging	NM	NM	NM	NM	Yes	Water-filled potholes	NED	NED	High	High power	Manageable	No
Aparna et. al	Thermal Imaging	95.2	NM	NM	NM	Yes	Not enough test information	Efficient	Yes	Manageable	Low	Manageable	Yes

NM: Not Mentioned

NR: Not Relevant

NED: Not Enough Data

2.4 Incline Detection and Assessment Techniques

Navigating the world presents a unique set of challenges for individuals who rely on wheelchairs for mobility. Seemingly ordinary features of the built environment, such as inclines and uneven terrain, can transform into significant obstacles. The absence of properly designed wheelchair ramps further exacerbates these difficulties, limiting access to public spaces and hindering social participation. These limitations have a profound impact on a person's sense of independence, dignity, and overall inclusion within their community.

Upward inclines pose a significant physical challenge for wheelchair users. The steeper the incline, the greater the exertion required to propel the wheelchair, leading to fatigue and potentially exceeding an individual's upper body strength. Downward inclines can be equally problematic, presenting difficulties in maintaining control and increasing the risk of accidental falls. Uneven surfaces further complicate navigation, creating instability and potentially causing damage to wheelchairs.

The lack of properly designed wheelchair ramps presents another critical barrier. Inadequate ramp slopes, excessively narrow widths, or missing handrails can render a ramp unusable or unsafe for individuals with varying mobility needs. Assessing whether the ramp is safe or not is not as easy as it may seem. A ramp can be considered untraversable and unsafe when its inclination is more than one degree higher than the standard set by the local regulators. This often results in forced detours, social exclusion, and a constant undercurrent of anxiety about encountering unexpected obstacles, as well as the challenges present when attempting to assess their traversability.

The limitations imposed by inclines and inadequate wheelchair ramps extend beyond manual navigation for individuals with disabilities. They also pose significant challenges for the ongoing development of autonomous and semi-autonomous vehicles. This section presents a critical analysis of existing techniques for addressing these challenges, offering a comprehensive and unbiased assessment of their performance, reliability, and scalability. To facilitate a rigorous comparison, a systematic evaluation of each technique's strengths and weaknesses will be conducted.

2.4.1 Review Strategy and Performance Measures

This section delves into a critical examination of existing techniques for detecting and localising upwards and downwards inclines. The review will encompass not only incline identification but also segmentation, which involves precisely delineating the incline's boundaries, and assessment, which refers to measuring its characteristics like inclination angle or slope. A core focus will be placed on understanding the underlying technological principles that drive each approach.

A rigorous evaluation framework will be established to provide a comprehensive comparison. This framework will assess the strengths and weaknesses of each method based on key design criteria crucial for real-world applications, particularly those involving detection and localisation. Here's a breakdown of these criteria:

- **Accuracy:** This criterion assesses how well a technique can correctly identify the presence of inclines and accurately measure their characteristics. Metrics such as accuracy and confidence will be employed to reflect the number of times a non-incline is mistakenly identified as one and will be employed from the research literature whenever available to quantify accuracy.

- **Precision:** This criterion focuses on how precisely a technique can locate and define the exact boundaries of an incline. For segmentation techniques that involve delineating incline borders and assessing the difference between the ground truth and the measured values.

- **Environmental Robustness:** Real-world environments present challenges like varying lighting conditions, weather fluctuations, and potential occlusions from objects or other terrain features. This criterion evaluates how well a technique maintains its performance under such complexities. For instance, techniques relying solely on vision might be more susceptible to lighting variations compared to those that incorporate depth sensors, which can provide more robust incline detection regardless of lighting conditions.

- **Computational Efficiency:** This criterion assesses the computational resources required by the technique. For applications such as autonomous navigation, real-time processing capabilities are essential. Techniques that rely on computationally intensive models might not be suitable for real-time deployment on resource-constrained platforms like autonomous vehicles.

By synthesising the insights gained from analysing the strengths and weaknesses of existing techniques, the goal is to pave the way for the development of a novel detection and localisation algorithm for inclines. This novel approach will aim to leverage the strengths of current methodologies, potentially by combining functionalities from different techniques, while offering superior accuracy and reliability for incline detection tasks, exceeding the limitations identified in the reviewed methods. Furthermore, the review will integrate relevant performance metrics like accuracy, error rate, processing time, and power consumption from the research literature to support the evaluation, providing a quantitative basis for assessing the effectiveness of existing techniques. Additional information can be found within Appendix B.

2.4.2 Vision or Radar-Based Systems

2.3.2.1 Point-Cloud-Based Techniques

Qiu et. al [79], and Heckman et. al [80] introduced a real-time algorithm for off-road terrain estimation using laser data that generates a point-cloud representation of the surrounding environment. addressing the challenges of complex calculation and object abstraction faced by the Gaussian mixture algorithm (GMA) in practical applications. The modified GMA introduces three key enhancements: a selection window based on the dominant-ellipse-principle to limit the probability distribution area, a clustering approach for efficient object distinction, and a virtual point vector to reduce the computational load of the mean square error matrix. These modifications improve the algorithm's real-time

performance and potential obstacle detection capability, as demonstrated through experiments on a tracked mobile robot.

Furthermore, Qiu et al. demonstrated the algorithm's efficiency in distinguishing obstacle contours and its computational speed compared to the ordinary GMA. Introducing the selection window, clustering strategy, and virtual point vector significantly enhances the algorithm's real-time feasibility and object-abstraction capability, addressing the limitations of the original GMA. The appendix provides a detailed explanation of the derivation process for the virtual point vector, offering a comprehensive insight into the algorithm's technical intricacies and implementation.

As per the introduced criteria, the system can be assessed as follows:

- **Accuracy:** The system's accuracy remains indeterminate due to a paucity of test cases. The reliance on simulated data limits the generalisability of performance claims.
- **Precision:** The lack of comprehensive metric data prevents a conclusive assessment of the system's precision. Further quantitative analysis is required.
- **Environmental Robustness**: Limitations inherent to the chosen sensing technique raise concerns about real-world implementation. Robustness cannot be verified without empirical testing in diverse environmental conditions.
- **Computational Efficiency:** The system exhibits real-time functionality, evidenced by a response time of 42.4531 ms. This suggests computational efficiency.

Zhend et. al [81] presented a method for real-time slope detection of planetary surfaces in the context of autonomous obstacle avoidance for planetary rovers via the use of LiDAR sensing.

The system is based on a slope detection method involving the down-sampling of original point clouds, voxel mesh filtering, 3D point cloud clipping and segmentation, local ground plane fitting estimation, and the classification of slopes based on predetermined thresholds. The system utilises sensor fusion by carrying different onboard sensors, including passive sensor measurement and active sensor measurement, and by using its multi-sensor fusion measurement technique, real-time results are obtained. Slope

estimation methods proposed include the Kalman filtering method, least square method, edge slope detection method, 3D point cloud fast segmentation method, data acquisition method of slope detection based on Trimble GX 3D laser scanner, and LiDAR method.

By employing the performance criteria discussed, the system can be assessed as follows:

- **Accuracy:** The limited number of test cases renders the system's accuracy indeterminate. Over-reliance on simulated data restricts the extrapolation of performance claims to real-world scenarios.
- **Precision:** The absence of comprehensive metric data impedes a definitive evaluation of the system's precision. Robust quantitative analysis is essential for an informed assessment.
- **Environmental Robustness:** Constraints inherent to the chosen sensing technique necessitate empirical testing in diverse environmental conditions to ascertain real-world robustness.
- **Computational Efficiency:** While the experiment suggests real-time functionality, further analysis is required to establish the consistency and computational efficiency of the system under varying workloads.

Meng et al. [82] presented a slope detection method tailored for quadruped robots equipped with 3D LiDAR sensors.

The proposed slope detection method leverages bilateral filtering and Random Sample Consensus (RANSAC) algorithms to analyse point cloud data obtained from the 3D LiDAR sensor. By fitting planes to the data using RANSAC, the method can accurately detect slopes in the environment. The RANSAC algorithm iteratively identifies inliers that fit the plane model while excluding outliers, ultimately estimating the slope angle and position. This approach allows for robust slope detection even in the presence of noisy data or outliers, enhancing the reliability of the results.

As per the criteria discussed, the system can be assessed as below:

- **Accuracy:** Observed detection rates suggest acceptable accuracy, however, additional rigorous testing is required for a comprehensive evaluation.
- **Precision:** The system exhibits precision within the acceptable low-end expected range (± 2 degrees), though a broader range of measurements would strengthen this claim.

- **Environmental Robustness:** Inherent constraints of the sensing technique raise concerns about the system's robustness across diverse environmental conditions. Empirical testing in variable environments is essential.
- **Computational Efficiency:** The experiment indicates real-time functionality, but further analysis under diverse workloads is needed to confirm consistency and computational efficiency.

Rusu et al. [83], Gutmann et. al. [84], and Snigh et al. [85] presented a stereo-vision technique, later improved by Murarka et. al [86, 87] presented a stereo vision-based mapping algorithm fused with image segmentation, and motion cues, focusing on enabling wheeled mobile robots to navigate safely in urban environments by detecting inclines, drop-offs, and obstacles

The algorithm segments the 3D grid into potentially traversable ground regions and fits planes to these segments using linear least squares, distinguishing between level, inclined, and non-traversable areas.

By applying the discussed criteria, the system can be assessed as follows:

- **Accuracy:** Even though the test results show a promising accuracy with the True Positive and True Negative rate being high (90%+), the results remain acceptable as this should be improved for the system to be usable by the public.
- **Precision:** The precision of the system is not addressed as the system provides w binary solution of the problem without a precise measurement of the inclination angle and the width of the plane.
- **Environmental Robustness:** The inherent characteristics of the sensing technique suggest potential vulnerabilities to varying environmental conditions, particularly changes in lighting. Rigorous empirical testing across diverse scenarios is crucial to determine the system's true robustness.
- **Computational Efficiency:** Whilst the experimental results imply real-time capability, comprehensive testing under varied workloads is necessary to confirm both the consistency of the system's real-time performance and its overall computational efficiency.

Brossette et al. [88], Cockrell et al. [89], and Tseng et. al. [90] presented an RGB-D-based algorithm for ramp and incline detection, which was later improved by Nejati et al. [91] whom presented an algorithm for automated real-time ramp detection using 3D point cloud data for powered wheelchairs, focusing on addressing the challenges faced by

individuals with severe motor impairments using binary control interfaces. The algorithm uses point cloud data collected from an RGB-D sensor to detect the orientation, slope, and width of traversable ramps without visual fiducial requirements and environmental customisation.

Based on the specified criteria, the system can be assessed as follows:

- **Accuracy:** The system's accuracy raises concerns as the average accuracy is only limited to 87.9% which prevents it from being real-world usable.
- **Precision:** The system provides precise measurements when the outcome is a True Positive. However, this remains limited to the region-growing algorithm's performance as false readings as possible due to overlapping objects
- **Environmental Robustness:** The inherent characteristics of the sensing technique suggest acceptable performance in real-world scenarios if the accuracy of the detection is sufficient. RGB-D image sensors are considered reliable in most cases.
- **Computational Efficiency:** Whilst the experimental results imply real-time capability, the system is at risk of inadequate performance due to the fact that the region-growing algorithm is the chosen segmentation algorithm. The chosen technique is also computationally expensive which is another issue that needs to be addressed.

2.3.2.2 Non-Point-Cloud-Based Techniques

J. Wu et. al [92] proposed implementation of Convolutional Neural Networks (CNN) to identify buildings with ramp entrances, as required by the Americans with Disabilities Act (ADA) to ensure equal access for individuals with disabilities. The aim is to develop an artificially intelligent system capable of classifying building images to determine the presence or absence of a ramp at the entrance, utilising CNNs for image classification.

By applying the discussed criteria, the system can be assessed as follows:

- **Accuracy:** The system's result reflect a reliable accuracy as it reaches 95.6%. This means that the accuracy is within the acceptable ranges.

- **Precision:** The system does not provide any measurements of the inclination angle and width of the incline plane which limits its capability in terms of precision.
- **Environmental Robustness:** The inherent characteristics of the sensing technique pose a limitation to the system's performance and usability as it cannot perform adequately in low-light scenarios.
- **Computational Efficiency:** The system utilises a CNN in order to perform its core detection and classification functions, hence, it requires computational power even though it can perform in real-time.

B. Wu et. al [93] presented an uphill safety controller for intelligent wheelchairs, incorporating deep learning-based ramp detection via the use of a CNN. The controller system consists of user buttons, ramp classes, and real-time camera screens. The controller algorithm is divided into two parts: fuzzy rule calculation of target speed based on user input and current speed and Q-learning-based ANFIS controller for computing the final output speed. The system employs a voting system and gyroscope data assistance to improve accuracy, resulting in a higher classification rate of ramp detection.

Based on the presented criteria, the system can be assessed as follows:

- **Accuracy:** The system's result reflects a reliable detection of 92.1%. This means that the accuracy is within the acceptable ranges.
- **Precision:** The system does not provide detailed measurements of the inclination angle of the detected planes as estimated by the system making the data available below the needed requirements.
- **Environmental Robustness:** The intrinsic limitations of the chosen sensing technique compromise the system's performance and practical utility, particularly in low-light and limited-visibility environments. This dependence on specific environmental conditions hinders reliable operation in diverse real-world settings.
- **Computational Efficiency:** The system's utilisation of a CNN for core detection and classification tasks, while enabling real-time performance, necessitates significant computational resources. This trade-off highlights the inherent tension between accuracy, often achieved by computationally complex models, and the constraints of real-time operation.

Lutz et al. [94] proposed a technique enabling a humanoid robot, specifically the Nao robot, to autonomously navigate ramps using only vision and inertial data. By integrating monocular vision and inertial measurements, the Nao robot successfully walks down a 2.10 m long ramp inclined at 20 degrees. The approach involves recognising the boundaries of the ramp by detecting the top and bottom edges using image processing techniques like the Canny edge detection algorithm and the probabilistic Hough transform. This allows the robot to carefully enter and exit the ramp without falling.

Based on the specified criteria, the system can be assessed as follows:

- **Accuracy:** Information given about the experiment is not sufficient to make this judgement.
- **Precision:** The system does not take into account the pre-measurement of the inclination angle of the ramp, which means that precision cannot be measured.
- **Environmental Robustness:** The intrinsic limitations of the chosen sensing technique limit the system's performance due to problems such as ambiguous scale and low-light scenarios.
- **Computational Efficiency:** The system was used on the NAO robot's internal processor, which means that it is not very computationally expensive, it can perform in real-time, and can be mounted onto an EPW.

2.4.3 Non-Vision or Radar-Based Systems

Tareen et al. [95] proposed two low-cost slope detection and calculation techniques using ultrasonic range finders, implemented on a robot named "The ROBUST," enabling it to differentiate between objects and sloppy paths. The first technique, Technique A involves fixed ultrasonic range finders and is limited in its applicability to a specific range of inclination/declination angles due to the deflection of ultrasonic waves from smooth surfaces. In contrast, Technique B, which involves a rotatable platform, offers more versatile and remote sensing of different orientations of slopes, including ditch detection and avoidance. The discussion also addresses the challenges and limitations associated

with ultrasonic range finders, including their susceptibility to interference, environmental conditions, and the need for signal conditioning to fine-tune the sensors for accurate slope detection.

By applying the discussed criteria, the system can be assessed as follows:

- **Accuracy:** The system's accuracy is not reliable due to the many limitation to the detection rate caused by the inherent limitations of the sensor technology used.
- **Precision:** The system, although not reliable to succeed in every detection, has an acceptable precision with a very low error margin.
- **Environmental Robustness:** The intrinsic limitations of the chosen sensing technique render the system unreliable due to the high risk of failure that ultrasound sensors face.
- **Computational Efficiency:** The system is not computationally expensive and therefore can be run in real-time without issues.

Yu et al. [96] proposed a technique for dead reckoning of a mobile robot in complex terrain based on proprioceptive sensors such as fibre optic gyro and tilt sensors. It emphasises the importance of accurate and reliable dead reckoning for mobile robot navigation and proposes a method that integrates multiple proprioceptive sensors' information to estimate the relative motion trajectory of a mobile robot. The proposed method involves the analysis of the kinematic model of a mobile robot and suggests the integration of the wheel-ground contact angle for more accurate estimation. The paper also discusses the dead reckoning algorithm, outlining the steps involved in sensor data pretreatment, rigid-body transformation, estimation of wheel-ground contact angles, calculation of velocity components, and the update of dead reckoning. Furthermore, the document presents the results of motion simulation and experimental analysis conducted to validate the proposed dead reckoning method using an experimental platform of a mobile robot with two rocker-bogie suspensions and four drive wheels.

By applying the discussed criteria, the system can be assessed as follows:

- **Accuracy:** The system exhibits limited accuracy in incline detection. This stems from inherent limitations within the chosen sensor technology, restricting the achievable detection rate.
- **Precision:** the system does not explicitly incorporate inclination measurement, precluding a detailed analysis of its precision in this regard.
- **Environmental Robustness:** The system demonstrates reduced environmental robustness due to the intrinsic limitations of the chosen sensing methodology. Specifically, IMUs are known to suffer from accuracy degradation and potential failure under specific environmental conditions.
- **Computational Efficiency:** The system demonstrates favourable computational efficiency, enabling real-time operation without significant computational overhead.

Zhiblin et al. [97] proposed an integrated control framework designed to enhance humanoid robot stability on uneven terrain. This framework seamlessly merges stabilisation control with terrain inclination estimation. Stabilisation is realised through passivity-based admittance control, leveraging force/torque feedback within the feet to modulate compliance actively. Terrain estimation is conducted via a logic-based algorithm that employs the feet as probes to determine surface inclination. This algorithm effectively manages under-actuation phases from foot tilting on the contact surface. The admittance controller dynamically adjusts equilibrium positions to facilitate balance recovery on sloped surfaces. As most robotic systems utilise position-controlled actuators, the study investigates an c that exploits force/torque feedback in the feet to alter joint position references, thereby promoting passivity substantiate the capabilities of the framework in terrain inclination estimation, slope-based balance adaptation, and upright posture maintenance.

As per the assessment criteria proposed, the system's performance is evaluated as below:

- **Accuracy:** The system exhibits limited accuracy in incline detection. This limitation arises from constraints inherent to the chosen sensor technology, restricting the achievable detection rate.
- **Precision:** The system does not explicitly address inclination measurement, precluding a detailed analysis of its precision in this respect.

- **Environmental Robustness:** The system demonstrates reduced environmental robustness due to intrinsic limitations of the chosen sensing methodology. Specifically, the contact exploration technique employed is known to suffer from accuracy degradation and potential failure under specific environmental conditions.
- **Computational Efficiency:** The system possesses favourable computational efficiency, facilitating real-time operation without imposing significant computational overhead.

In order to provide a comparative framework for the discussed techniques, the table below is assimilated:

Table 2.4: **Incline detection and assessment techniques comparison table**

Authors	Sensory Method	Main Limitation	Accuracy	Precision	Environmental Robustness	Computational Efficiency
Qiu et al.	Laser Vision	Environmental Sensitivity	NED	NED	Limited	Real-Time Detection
Zheng et al.	LiDAR Sensing	Environmental Sensitivity	NED	NED	Limited	NED
Mend et al.	LiDAR Sensing	Environmental Sensitivity	Acceptable	Acceptable	Limited	Real-Time Detection
Murarka et al.	Stereo Vision	Binary Solution	Acceptable	NM	NED	Real-Time Detection
Nejati et al.	RGB-D Sensor	Computational Power	Average	High	Acceptable	NED
J. Wu et al.	RGB Sensor	Binary Solution	Acceptable	NM	Limited	Real-Time Detection
B Wu et al.	RGB Sensor	Sensor Reliability	Acceptable	NM	Limited	Real-Time Detection
Lutz et al.	Monocular Vision + IMU	Sensor Reliability	NED	NM	Limited	Real-Time Detection
Tareen et al.	Ultrasonic Sensors	Sensor Reliability	Not Reliable	Acceptable	Limited	Real-Time Detection
Yu et al.	Proprioceptive Sensors	Sensor Reliability	Limited	NM	Limited	Real-Time Detection
Zhiblin et al.	Admittance Controller	Sensor Reliability	Limited	NM	Limited	Real-Time Detection

NM: Not Mentioned

NR: Not Relevant

NED: Not Enough Data

2.5 Downwards-facing Stairs and Curbs Detection and Localisation Techniques

Individuals with mobility impairments often rely on electric-powered wheelchairs (EPWs) and other assistive technologies (ATs) to navigate their daily lives, to them, every day poses a new challenge to the previous with numerous obstacles and limitations. Despite the advancements in these technologies, users continue to face a multitude of challenges that hinder their independence, inclusion, and overall quality of life. These challenges stem from limitations within the built environment, societal attitudes, and the inherent constraints of current assistive devices.

The built environment presents numerous obstacles for EPW and AT users. Architectural barriers such as uneven terrain, lack of proper wheelchair-accessible curbs and their cuts, the lack of proper safety markings to warn from downwards-facing stairs, and other everyday challenges severely restrict mobility and access to essential spaces. Social stigma and negative assumptions about ability can create further barriers, leading to discrimination and reduced participation in social activities. Additionally, while ATs offer significant benefits, they can have limitations regarding manoeuvrability, adaptability to diverse environments, and ease of use.

Downwards-facing stairs and curbs present particularly hazardous obstacles, introducing risks of falls, collisions, and a significant curtailment of independent mobility. The development of reliable and accurate downwards-facing stairs and curb detection and localisation techniques is crucial for enhancing the safety, autonomy, and overall quality of life for EPW and AT users.

Existing literature offers a range of approaches to address this challenge. Traditional computer vision techniques often rely on geometric feature extraction and pattern recognition. These methods can be computationally efficient but may struggle with adaptability under varying lighting conditions or complex environments. Alternatively, sensor-based solutions using ultrasonic sensors, depth cameras, or LiDAR can provide more robust depth perception. The recent surge in machine learning, particularly deep learning, has driven innovation in this field. Convolutional neural networks (CNNs) have demonstrated promising results in image-based obstacle classification, including stairs and curbs. However, challenges persist in dataset acquisition, the need for large computational resources, and ensuring system generalisation across diverse real-world scenarios.

Beyond obstacle detection, accurate localisation is essential for safe and effective navigation by EPWs and ATs. Research in this area explores techniques such as sensor fusion, simultaneous localisation and mapping (SLAM) algorithms, and the integration of GPS data. These approaches aim to provide precise real-time information about the position and orientation of obstacles relative to the assistive device.

2.5.1 Review Strategy and Performance Measures

This section critically examines the state-of-the-art techniques for downwards-facing stairs, curb detection, and localisation specifically designed for assistive technologies and EPWs. It will analyse the strengths and limitations of existing methodologies across a range of parameters, including accuracy, reliability, computational efficiency, adaptability, and

cost-effectiveness. Additionally, the review will identify key research gaps and potential directions for future innovation.

A robust evaluation framework will be established to facilitate a rigorous comparative analysis of incline detection and localisation methods. This framework will systematically assess the merits and limitations of each technique according to core design criteria essential for real-world deployment. Key criteria include:

- **Accuracy:** The technique's ability to correctly classify curbs and downward-facing stairs while accurately measuring their dimensions. Metrics such as classification accuracy (percentage of correct classifications) and confidence scores will be used to quantify accuracy.
- **Precision:** The technique's ability to precisely delineate the boundaries of a traversable path, taking into account the spatial relationship between curbs, downward-facing stairs, and the path itself. Segmentation techniques will be evaluated using metrics that compare the deviation between predicted boundaries and ground truth.
- **Environmental Robustness:** The technique's performance under real-world complexities such as varying lighting, weather conditions, and potential occlusions. This evaluation will consider the potential benefits of multi-sensor fusion (e.g., combining vision and depth sensors) for enhanced resilience compared to vision-only approaches.
- **Computational Efficiency:** An assessment of the computational resources required by each technique. Real-time path planning for autonomous navigation demands efficient algorithms. Techniques with high computational overhead might necessitate specialised hardware or be unsuitable for resource-constrained platforms.

A critical synthesis of the strengths and weaknesses identified within existing curbs and downwards-facing stairs detection and localisation techniques will serve as the foundation for the development of a novel algorithm. This novel approach will strategically

leverage the identified advantages of current methodologies. One potential approach could involve the fusion of functionalities from complementary techniques. The primary objective will be to surpass the limitations of reviewed methods by achieving superior accuracy, precision, and robustness in incline detection tasks.

The proposed evaluation framework will play a pivotal role in this development process. By integrating relevant performance metrics from the research literature (e.g., accuracy, error rate, processing time, power consumption), the framework will provide a quantitative basis for objectively assessing the effectiveness of existing techniques. This comparative analysis will serve as a guiding principle for strategically combining the strengths of existing methodologies and ultimately inform the design of the novel algorithm. Additional information can be found within Appendix C.

2.5.2 Vision or Radar-Based Systems

2.3.2.1 Point-Cloud-Based Techniques

Ashraf et al. [98] introduced a system that performs the autonomous detection of stair dimensions for motion planning of stair-climbing robots. . The proposed approach uses a depth camera to determine stair features necessary for autonomous motion planning. It involves computing the point cloud of the perceived environment using a pinhole camera model and organising the data into horizontal and vertical clusters to generate classes for each step. The Maximum Likelihood Estimation Sample Consensus (MLESAC) algorithm is used to fit a plane on each class to accurately determine stair dimensions. Various existing methods using sensors like monocular cameras, stereo cameras, and LIDAR are also briefly discussed, highlighting their advantages and limitations. The proposed approach with a depth camera is described as accurate, low-cost, and suitable for real-time operation with high precision

As per the discussed performance assessment criteria, the system can be evaluated as follows:

- **Accuracy:** The system achieved an average accuracy of 93.63% when tested in a simulation environment which suggests an acceptable accuracy.
- **Precision:** The detection precision is not mentioned within the experimental results which suggests that further testing in required.
- **Environmental Robustness:** The detection method's inherent limitations form a serious limitation to the system. Mainly the issue of the field of view, and the low-light performance.
- **Computational Efficiency:** The system is computationally expensive, which means that it not the answer for a real-time detection algorithm.

Apellániz et al. [99] proposed a technique for detecting 3D curbs in point cloud data from LiDAR sensors through a process involving the use of deep neural networks and odometry information. The detections generated can serve as pre-annotations for annotation tools, reducing manual annotation time by about 50%. The methodology includes scan-level curb detection using a DNN, transforming 2D predictions to 3D curb points, reconstructing 3D curbs for entire sequences, and post-processing steps like clustering, skeletonisation, and simplification to generate final polylines. The goal is to provide standardised curb annotations that can be utilised in annotation tools efficiently.

By applying the discussed criteria, the system can be assessed as follows:

- **Accuracy:** The system exhibits an acceptable accuracy of 87.8%. This limits its usability in assistive technologies, however, it remains a proportionally high rate.
- **Precision:** The system's precision peaks at 90.7% making it acceptably precise.
- **Environmental Robustness:** The inherent limitations caused by the chosen sensing technique (LiDAR) render the system susceptible to environmental challenges.
- **Computational Efficiency:** The system requires a significantly high computational power making it a less favourable candidate for real-time detection tasks.

Zhao et al. [100], Hata et al. [101], Zhang et al. [102, 103], Huang et al. [104], Horváth et al. [105] , Yao et. al. [106], Yu et al. [107], Wang et al. [108], Guo et al. [109] , Zhu et al. [110], Jung et al [111], and Yamamoto et al. [112] utilised the LiDAR technology to detect curbs. This was also used and improved by Gurrero et al. [113] who presented a technique for road curb detection using a LIDAR 3D sensor in urban environments for autonomous vehicle technology. The method includes ground segmentation, geometrical feature extraction, and the utilisation of LIDAR 3D reflectance features for curb detection.

The system includes methods for 2D translation and yaw angle determination using calibration patterns on an autonomous shuttle. It utilises ground segmentation techniques using LIDAR data to detect road curbs, emphasising the importance of accurate segmentation in detecting road boundaries. The system also includes an algorithm for ground segmentation based on specific criteria and an approach for road curb detection with consideration of geometric features and LiDAR reflectance data. Experimental tests were conducted to validate the road curb detection method on an autonomous shuttle vehicle. Additionally, reflectance data is used to improve the precision and sensitivity of road curb detection algorithms.

As per the mentioned criteria, the system can be assessed as follows:

- **Accuracy:** The system's accuracy (83.34%) falls below the minimum expectations for real-time detection performed whilst being mounted onto an electric-powered wheelchair (EPW)
- **Precision:** The system's precision is acceptably high (90.56%). Even though it falls below expectations, it can still be used for less critical systems.
- **Environmental Robustness:** The inherent limitations of the LiDAR sensor prevent the technique from being considered environmentally robust
- **Computational Efficiency:** The system requires high computational power which brings it to being less favourable when it comes to real-time detection.

2.3.2.2 Non-Point-Cloud-Based Techniques

Zou et al. [114] chose deep learning, a technique previously used by Patil et al. [115] to detect stairs. Zou et al.'s method performs road curb detection using a deep learning framework, particularly focusing on detecting curbs in single road images for autonomous driving. The method utilises a customised convolutional neural network (CNN) based on ResNet18 for feature extraction to classify curb and no-curb regions within a boundary box.

By employing the mentioned criteria, the system can be assessed as follows:

- **Accuracy:** The system's reported accuracy is 99.4%. However, there remains concerns about its validity.
- **Precision:** The system's reported precision is 100%. However, concerns are raised as this is not illustrated in the provided test results.
- **Environmental Robustness:** The inherent limitations of the RGB sensor render the technique environmentally unrobust .
- **Computational Efficiency:** The system requires high computational power which brings it to being less favourable when it comes to real-time detection.

Zhou et al. [116] presented LaCNet, a network dedicated to real-time detection of arbitrary-shaped lanes and curbs through instance segmentation. This network combines lane and curb detection tasks, addressing challenges such as differentiating various instances with tiny gaps and complex spatial relationships encountered in real driving scenarios.

As per the discussed criteria, the system can be assessed as follows:

- **Accuracy:** The system's accuracy is represented by its F1-Score 98.65% which is a notable result.
- **Precision:** The system's reported precision is 98.42%, which is a notable result.
- **Environmental Robustness:** The inherent limitations of the RGB sensor render the technique environmentally unrobust .

- **Computational Efficiency:** The system performs in real-time. Howeverm it requires high computational power making it unsuitable for assistive technologies.

Mihankhah et al. [117] utilised laser sensors to detect stairs in real time. This was later used by Byun et al. [118] to detect curbs, and improved by Pollard et al. [119] who proposed a system for step and curb detection using laser sensors for Personal Mobility Vehicles (PMV) in urban areas where steps and curbs can be barriers for individuals with mobility issues. The system is based on an algebraic derivative method applied to laser sensor data to detect steps in front of the vehicle.

With the help of the mentioned criteria, the system can be assessed as follows:

- **Accuracy:** The accuracy of the system was not reported. Although the resultant experiment shows an acceptable accuracy.
- **Precision:** The reported precision of the system is 98.42%.
- **Environmental Robustness:** The inherent limitations of the RGB sensor render the technique environmentally unstable .
- **Computational Efficiency:** The system requires high computational power which brings it to being less favourable when it comes to real-time detection.

Manuel et al. [120], Cai et al. [121] and Panev et al. [122] attempted to use an RGB camera to tackle the problem. However, Panev et. al used the monocular forward-view camera already fitted on many cars as a standard. This camera is usually fitted with a fisheye lens. He proposed method combines 3-D geometric reasoning with advanced vision-based detection methods to estimate the size, location, and orientation of road curbs. This technology is crucial for advanced driver assistance systems to prevent collisions and damages during parking manoeuvres. The system involves curb detection in individual video frames and temporal analysis for false positive rejection.

By applying the assessment criteria discussed before, the system can be evaluated as follows:

- **Accuracy:** The accuracy of the system, as reported (91.4%) is acceptable
- **Precision:** The reported precision of the system is represented by the F1-Score and calculated as 92.3%, making it acceptable.
- **Environmental Robustness:** The inherent limitations of the RGB sensor render the technique environmentally unstable. The reported testing weather conditions only account for clear and cloudy weather.
- **Computational Efficiency:** The system requires high computational power which brings it to being less favourable when it comes to real-time detection.

Cheng et al. [123] introduced a curb detection for road and sidewalk detection in urban residential areas using stereo vision. The method involves estimating flat areas, creating a 16-dimensional descriptor for curbs, employing a vanishing point-constrained model for road detection, utilising a region-growing method for sidewalk detection, and implementing a classification framework with Support Vector Machines (SVM) for curb point detection. Curb points are detected using a classification framework with a Support Vector Machine (SVM) model and how false positives are removed.

By employing the discussed evaluation criteria, the system is assessed as follows:

- **Accuracy:** The accuracy of the system is represented by its maximum F1-Score of 95.27%, which makes it considerably acceptable.
- **Precision:** The reported precision of the system 94.57%, making it notable.
- **Environmental Robustness:** The inherent limitations of the segmentation and classification techniques make it environmentally unstable.
- **Computational Efficiency:** The system requires high computational power which brings it to being less favourable when it comes to real-time detection.

2.5.3 Non-Vision or Radar-Based Systems

Rhee et al. [124] utilised ultrasonic sensors for curb detection whilst Bouhamed et al. [125], Razavi et al. [126] utilised them for staircase detection. This was improved later by Hatua et al. [127] who introduced a system implemented to detect universal staircases of dimensions

within a specific standard. The algorithm for stair detection is developed based on a comparative study of different dimensions of stairs. The ultrasonic sensor emits sound pulses to compute distances to the target object.

To assess the system, the evaluation criteria is applied:

- **Accuracy:** Not enough data has been supplied to assess the accuracy of the system.
- **Precision:** Not enough data has been supplied to assess the precision of the system.
- **Environmental Robustness:** The inherent limitations of the used techniques make it environmentally unstable.
- **Computational Efficiency:** The system requires low computational power which means that it can be used in real-time scenarios.

In order to provide a comparative framework for the discussed techniques, the table below is assimilated:

Table 2.5: **Stairs and curbs detection and localisation techniques comparison table.**

Authors	Sensory Method	Main Limitation	Accuracy	Precision	Environmental Robustness	Computational Efficiency
Ashraf et al.	Depth Camera & Point Cloud	Scope of the detection	Acceptable	NM	Limited	Not Real-Time
Apellániz et al.	LiDAR Sensing & DNN	Environmental Sensitivity	Acceptable	Acceptable	Limited	Not Real-Time
Gurrero et al.	LiDAR Sensing	Environmental Sensitivity	Insufficient	Acceptable	Limited	Real-Time Detection
Zou et al.	RGB & CNN	Detection Only	Acceptable	Acceptable	Limited	Not Real-Time
Zhou et al.	RGB & CNN	Environmental Sensitivity	High	High	Limited	Real-Time, High Power Consumption
Pollard et al.	Laser Sensor	Sensor Reliability	NM	High	Limited	Not Real-Time
Panev et al.	Monocular Vision	Sensor Reliability	Acceptable	Acceptable	Limited	Real-Time, High Power Consumption
Cheng et al.	Stereo Vision	Sensor Reliability	High	High	Limited	Real-Time, High Power Consumption

Rhee et al.	Ultrasonic Sensors	Sensor Reliability	NED	NED	Limited	Real-Time Detection

NM: Not Mentioned

NR: Not Relevant

NED: Not Enough Data

2.6 Discussion and Conclusion

A review of existing literature unveils a heavy emphasis on machine vision for obstacle detection in autonomous, semi-autonomous, and manually driven vehicles and assistive technologies. However, each method suffers from limitations that could endanger users, rendering them fallacious for real-time navigation. Achieving a truly autonomous obstacle avoidance remains a significant challenge due to the complexities of the environment. Unpredictable variations in pavements and diverse curb configurations present difficulties for detection algorithms. Additionally, the deceptive similarity between wheelchair ramps and potentially hazardous bicycle ramps poses a significant risk for assistive technologies. In addition to the previous, the irregular presence and varied shapes of potholes and cracks, potentially filled with water, ice, or reflecting bright light, rastically preclude the performance of individual sensor systems.

These environmental elements highlight the limitations inherent in singular sensor modalities. RGB cameras struggle with water, ice, low light, and strong light. Thermal cameras are hampered via high temperatures. Reflective lasers are susceptible to reflections caused by water and ice. LiDAR sensors face limitations in weather sensitivity and handling reflective surfaces. Processing power and energy consumption pose further constraints. Some systems require intensive computational resources, while others demand substantial power to operate sensors and processors.

Real-time functionality is important for safe navigation. Detection should be near-instant for accurate obstacle avoidance or safe path traversal, a must that is not met by current systems. Finally, the overall system size can be a limiting factor, with bulky equipment or large power sources incompatible with assistive technologies' size and weight constraints. Whilst design optimisation can often address this, it remains a

significant consideration. Mass production feasibility is another factor. Although manufacturability is generally achievable, the final system cost might be a significant barrier to universal adoption.

In conclusion, the limitations associated with individual obstacle detection and safe path traversal techniques necessitate a shift towards a multimodal approach. By intelligently combining various sensor modalities and robust processing algorithms, we can strive to achieve a more comprehensive and reliable obstacle detection system for autonomous vehicles, ultimately enhancing safety and paving the way for the widespread adoption of this technology.

In order to seamlessly overcome the numerous challenges posed, different approaches are to be taken for every obstacle's case:

- **Negative Road Anomalies:** This challenge will be overcome by implementing an algorithm that fuses data obtained using a CNN trained to detect potholes accurately. To accurately localise the detected objects, data from a depth sensor will be fused with the RGB sensor's data to add spatial understanding to the algorithm. The CNN detector to be chosen must be light and minimally complex to avoid computational power limitation.
- **Inclines and Wheelchair Ramps:** This challenge will be overcome by implementing an algorithm that uses point cloud data obtained via a depth sensor. This will reduce the system's power consumption and computational cost, enabling it to be used in conjunction with the detection of negative road anomalies and the localisation system. The point cloud data will be segmented, and the inclination angle of the candidate plane will be calculated. Its traversability will be assessed, taking into consideration its inclination angle and width, to ensure it is safely traversable.
- **Downwards-facing Stairs and Curbs:** This challenge will be overcome by implementing an algorithm that uses point cloud data obtained via a depth sensor. Instead of detecting downwards-facing stairs and curbs, and to reduce the complexity, the traversable area will be detected, segmented, and assessed. A safe distance will be determined to ensure that the user's mobility platform is kept at a safe distance from the edge of the traversable area.

Employing the presented methodologies enables a holistic system to be designed and implemented. This system would constitute a comprehensive novel framework that provides users and autonomous, or semi-autonomous navigation systems with sufficient crucial information enabling them to safely avoid negative obstacles, and to prudently traverse upwards and downwards-facing inclines and ramps.

Chapter 3

A Manually Collected Potholes Images Dataset

3.1 Introduction

Potholes present a significant challenge for autonomous systems, particularly assistive technologies like Electric-Powered Wheelchairs (EPWs) and mobility scooters. Their unpredictable shapes, variable reflectivity when water-filled, and potential for dangerous user falls highlight the need for robust detection solutions. Deep learning has emerged as a promising technique for pothole detection, offering the potential to adapt to diverse visual conditions. However, existing datasets are mainly focused on potholes observed from a dashcam or driver's perspective and are mostly directed towards on-road vehicles such as cars, vans, motorbikes, and other on-the-road vehicles. They also often lack sufficient representation of water-filled, debris-filled, and diversely coloured potholes. This limitation hinders model training for real-world scenarios. This dataset addresses this gap by providing 713 high-quality, manually collected images featuring 1152 manually annotated potholes following a pre-processing phase where images undergo a number of improvements before being processed within the labelling phase. Every image has been manually labelled via the use of an open-source labelling software "LabelIMG" [128]. Images were annotated with meticulous bounding boxes, ensuring maximum tangency to the target objects (potholes). This approach aimed to precisely constrain the objects while

simultaneously preserving sufficient background context for robust analysis of surrounding conditions. This UK-focused dataset emphasises diverse conditions, enhancing the potential for training robust detection models within real-world mobility scenarios. Additionally, the inclusion of images obtained from Kent County Council (KCC) adds valuable real-world data to complement the manually collected images. The proposed dataset has been publicly made available in [129], and extensively described within the already-published article [130].

This chapter is organised as follows: statement of problem is described in section 3.2. The data collection is discussed in section 3.3. The motivation is discussed in section 3.4, the manually-collected dataset is illustrated and described in section 3.5, and the chapter's conclusion, and future improvements are presented in section 3.6

3.2 Statement of Problem

The highly variable nature of potholes poses significant research and engineering challenges. Potholes, or surface depressions or failures, lack uniformity in their morphology and distribution. They may present diverse shapes, depths, locations, orientations, and textural characteristics. The presence of occluding materials further complicates the issue. This variability highlights the limitations of standardised detection techniques and necessitates innovative solutions. The challenges posed by potholes extend beyond detection, impacting both public safety and infrastructure integrity. They represent a considerable hazard to motorists, contributing to vehicular damage and potentially dangerous accidents. Traditional methods of manual inspection for potholes are notoriously inefficient and susceptible to human error. While automated detection systems offer promise, existing algorithms often struggle to accommodate the extensive variability in pothole manifestations. The lack of sufficiently diverse image datasets for pothole detection further compounds this problem, hindering the development and rigorous evaluation of machine learning models capable of accurately generalising to the complexities of real-world environments.

3.3 Data Collection

The dataset was constructed using images collected with a Samsung Galaxy Note 8 (13-megapixel camera). To ensure representativeness, image collection spanned various locations within Kent County, United Kingdom, including urban and rural roads and footpaths. Data was collected under diverse weather conditions (sunny, cloudy, and rainy) during both daytime and nighttime, enhancing the dataset's complexity. Potholes were photographed from four arbitrary heights and positions within a minimal distance of 50 centimeters, and a maximum distance of two meters. In addition, to the images taken via the phone sensor, a small number of images have been obtained through Kent County Council (KCC) who provided a small sample of pothole images submitted to the council within different claims. Collected images were resized to 412 x 412 pixels when possible, if not, either the height or width of the image was limited to 412 pixels (whichever possible). This enables the dataset to be versatile and usable in any deep learning network training scenario. On certain occasions, and in order to improve the understanding of the potholes' stochastic nature, images of the same pothole were taken from different angles and heights. This is to enable potential systems to be trained by using the dataset to have a clearer spatial understanding of the nature of the object to be detected.

3.4 Motivation

While publicly available datasets often contain substantial image quantities, many exhibit limitations stemming from their acquisition methods or content. Datasets derived exclusively from dashcam footage may offer limited viewpoints, while others lack diverse representations of pothole manifestations, including those filled with water, debris, or exhibiting irregular textures. Although models trained on these datasets may achieve acceptable performance on similar images, their detection capabilities become constrained when exposed to real-world scenarios with greater variability. This highlights the crucial need for a new, comprehensive dataset that encompasses the full spectrum of

pothole appearances. Such a dataset would be instrumental in developing robust detection algorithms adaptable to complex and less predictable environments.

As below, the dataset covers different cases of pothole instances, with additional cases not limited to what is showcased.

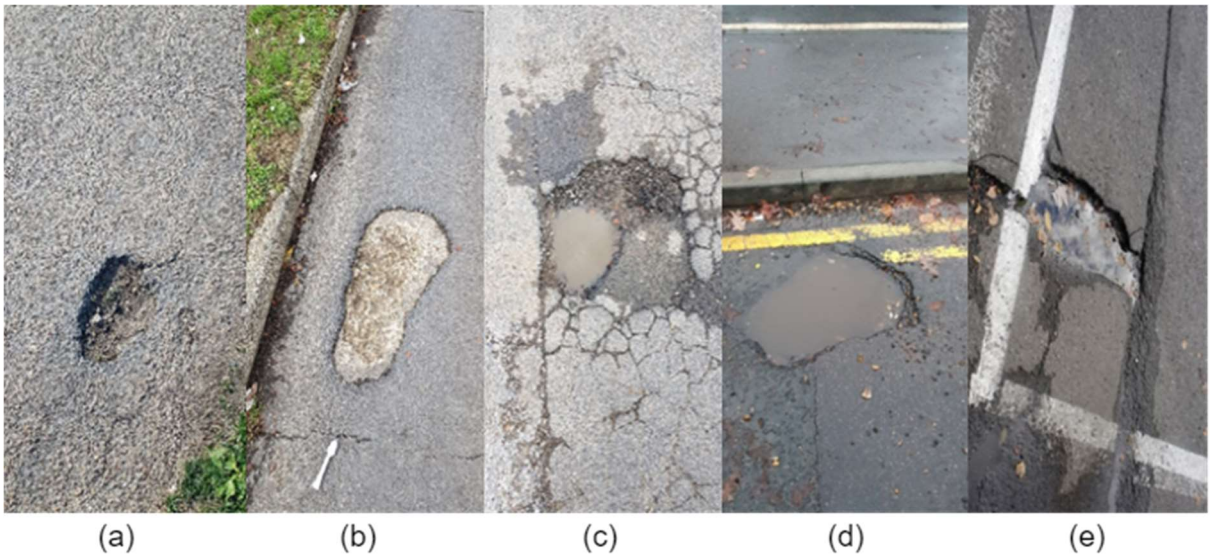


Figure 3.1: **Sample images taken from the manually-collected dataset**

Figure 1 illustrates a small sample of the potholes contained within the proposed dataset, these images can be described as below:

(a) Semi-circular Tarmac Depression: A classic pothole morphology, characterised by a semi-circular depression in the asphalt surface. This well-defined shape may be detectable by various systems unless occlusions are present.

(b) Shallow Surface Degradation: A widespread form of pavement deterioration, exhibiting surface cracking without significant depth. The lack of depth cues poses challenges for laser-based detection systems that rely on geometric profiling.

(c) Irregular Pothole with Occlusions: Presents a complex detection scenario due to its irregular shape and the presence of water, dirt, and debris. These occluding materials can interfere with laser, lidar, and sonar-based systems.

(d) Water-filled Pothole with Low Contrast: The near-transparent water and location near yellow lines creates a low-contrast environment, challenging image-based systems. Additionally, laser, sonar, and lidar systems may be affected by the reflective properties of water.

(e) Complex Occluded Pavement Failure: Marked by irregular shape, water, debris (rabble, leaves), and its location within a marked parking bay. This combination creates a highly challenging scenario for image processing, laser, lidar, and sonar-based detection due to reflections, lack of geometric cues, and the diverse nature of the occluding materials.

3.5 The Manually-Collected Dataset

The proposed dataset consists of 713 meticulously hand-annotated images, encompassing a total of 1157 potholes with their distribution visualised in Figure 3.2. A notable finding is the predominance of water-filled potholes within the dataset, addressing a critical gap in publicly available datasets which often lack sufficient representation of this challenging pothole type with its characteristic reflectivity and variable water clarity. Figure 3.3 further illustrates the distribution of potholes per image, revealing that the majority of images contain a single pothole, while a smaller proportion contain multiple potholes. This distribution is valuable for understanding the complexity of real-world detection scenarios. Additionally, analysis of the bounding box areas relative to the total image area allows for the calculation of average pothole-to-image ratios, providing quantitative insights into the relative size and density of potholes within the dataset.

To quantify the average spatial extent of potholes within the dataset, the ratio of each bounding box area to its respective image area is calculated. This yields the average pothole-to-image area ratio as defined by the equation:

$$Ratio = \sum \frac{Height_{(Bounding\ Box)} \times Width_{(Bounding\ Box)}}{Height_{(Image)} \times Width_{(Image)}} = 28.7529\ \% \qquad (3.1)$$

This means that the image dataset offers sufficient visual information to facilitate accurate pothole segmentation from the background, provided that the detection system adequately models the surrounding context.

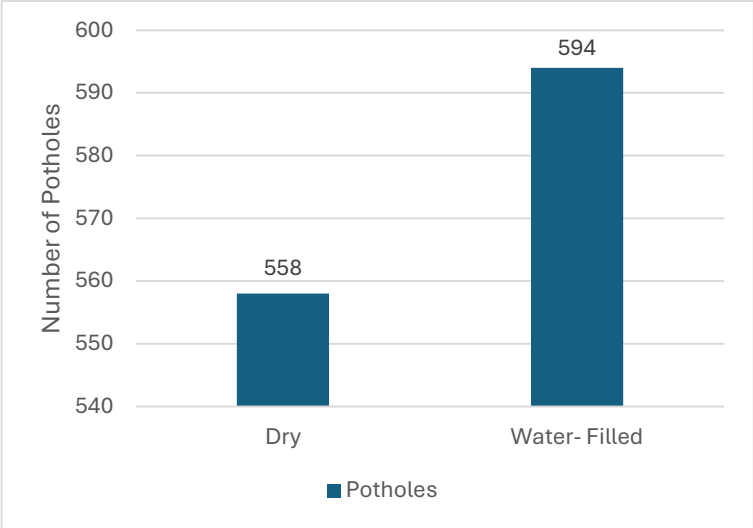


Figure 3.2: **Distribution of potholes in the proposed dataset**.

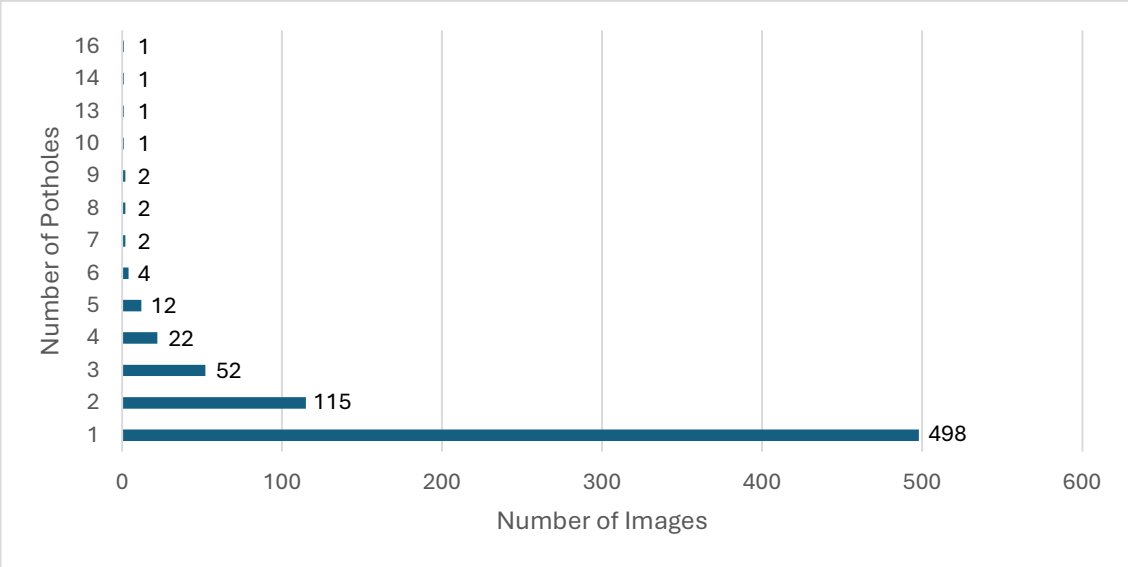


Figure 3.3: **Number of potholes in images**.

As a first step, the pre-processing stage took place whereby the collected images were downscaled to 30% of their original size. his targeted a width of approximately 415 pixels, ensuring compatibility with the input requirements of most contemporary, state-of-the-art object detection algorithms. Images were then individually annotated using LabelImg [128] an open-source Python labeling software. During the annotation process, bounding boxes were rigorously drawn to precisely enclose the pothole contours. This approach minimises the inclusion of extraneous background pixels within the bounding boxes (as shown in Figure 3.4), aiding in the prevention of divergence during the model training process.

Figure 3.4: **Image labelling as completed using the open-source software described in this section**. In the image, the pothole object has been surrounded by a manually-drawn bounding box labelled as "pothole". This will serve as an indication to the object detector during the training process.

The annotations were saved in two distinct formats. First, YOLO Darknet [131], a text-based format, was used. Each corresponding .TXT file (sharing the image's name) contains the following: the class number (in this case, a single class, "Pothole"), the bounding box coordinates (X1, Y1, X2, Y2), and the bounding box's height and width (as illustrated in Figure 3.5).

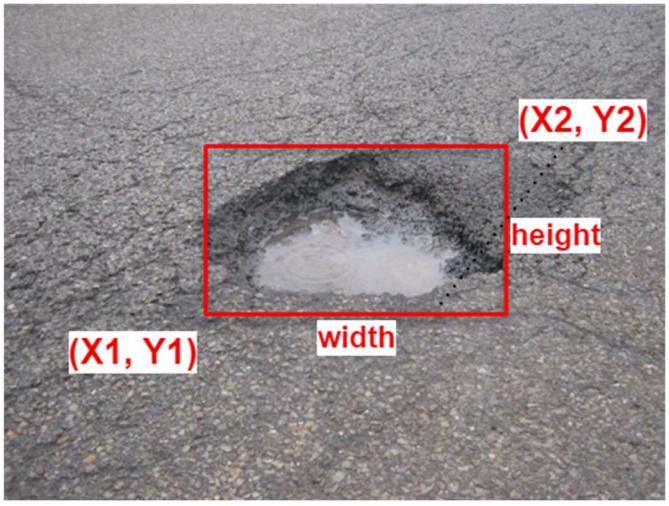


Figure 3.5: **YOLO Darknet Annotation Format.**

In addition to the YOLO Darknet format, images were annotated using the PASCAL VOC XML format, a widely recognised standard for object detection tasks. This format employs XML files that share the name of their corresponding image. Within the "<annotation>" tag, the file stores the image's file name, location, and dimensions (width, height, and depth). The object's bounding box is defined within the "<bndbox>" tag, using coordinates xmin, ymin, xmax, and ymax (equivalent to YOLO Darknet's x1, y1, x2, y2). Other tags, such as "truncated," "segmented," and "difficult," are set to 0 as they are inapplicable in this specific case. The PASCAL VOC format offers flexibility for future extensions; if necessary, researchers could incorporate these additional tags to describe more complex image characteristics. Figure 3.6 provides an illustrative example of a PASCAL VOC XML file.

```xml
<annotation>
  <folder>XML</folder>
  <filename>Pothole-022.jpg</filename>
  <source>
      <database>Pothole</database>
  </source>
  <size>
    <width>392</width>
    <height>806</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>pothole</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>67</xmin>
      <ymin>195</ymin>
      <xmax>287</xmax>
      <ymax>512</ymax>
```


Figure 3.6: **Pascal VOC XML Annotation Format**

The selected annotation formats—YOLO Darknet and PASCAL VOC XML—reflect their widespread adoption within popular object detection environments. This strategic choice empowers researchers to seamlessly integrate the proposed dataset into their work, facilitating the swift training of object detection models without the added burden of format conversions.

The proposed dataset offers raw data with minimal preprocessing, limited to resizing images to approximately 412 x 412 pixels (prioritising a 412-pixel width or height). This approach preserves flexibility for future developers, allowing them to tailor application methods and preprocessing techniques to the specific demands of their research. For training and validation, an 80/20 ratio was employed, with 80% of images randomly assigned for training and the remaining 20% for validation. We recommend replicating this random split after shuffling the images to avoid potential biases. The dataset's diverse backgrounds and objects, coupled with the relatively small average pothole-to-image area ratio (28.7529%), provide ample context for deep learning and AI algorithms, facilitating accurate pothole recognition across various conditions (time of day, weather, location). It's important to note that while the dataset offers a strong foundation, developers may need to consider additional augmentations (e.g., rotations, brightness adjustments) to further enhance algorithm robustness, particularly in challenging real-world scenarios.

3.6 Conclusion

The proposed dataset's diverse pothole conditions (shapes, light conditions, water-filled or dry) offer a crucial foundation for robust AI-based pothole detection systems, providing them with the extensive training data necessary to handle real-world variability. This comprehensive dataset holds significant value for researchers in computer vision, robotics, autonomous vehicles, and road safety, facilitating the development of systems that can reliably identify potholes in varied environments. Notably, it uniquely addresses both wet and dry potholes, addressing a critical gap in existing datasets. The dataset's potential applications are wide-ranging: from enhancing the navigation capabilities of autonomous vehicles and driverless cars to supporting path planning algorithms and enabling proactive pothole reporting for local authorities to optimse road maintenance. Additionally, the data has the potential to resolve limitations and challenges identified in previous research [132]. The dataset was collected the aim to fill the gap in modern publicly-available datasets. It contains a comprehensive amount of data covering many of the overlooked cases of pothole existence which can be an Achilles' heel to many AI-based object detection algorithms that lack the coverage of such neglected cases of stochastic existence of potholes within different shapes, textures, filling, and depth.

Future work within this dataset will focus on including more cases of ice-filled potholes as these are practically rare in the South-East England region, additionally, future annotation methods will include additional data representing an estimated depth of the pothole in order to enable potentially training a network that detects the depth of a pothole by simply detecting it within an RGB feed obtained via a nomal widely-available RGB-Camera.

Figure 3.7: **Supplemental dataset samples.** These samples demonstrate the range of real-world conditions covered within the dataset**.** The demonstrated samples illustrate different challenging scenarios of pothole instances ranging from potholes located within road markings to rabble-filled potholes, water-filled potholes, rabble and water-filled potholes, and shallow potholes.

Chapter 4

Negative Road Anomalies (Potholes) Detection and Localisation

4.1 Introduction

Negative Road Anomalies have long posed a significant challenge to the safety of both road and pavement users. Defined as irregularities or deviations from an ideal pavement or road surface, negative road anomalies entail potholes, cracks, uneven surfaces, rutting, and other road or pavement imperfections. Potholes, or depressions in road surfaces, are the universal and most general road anomalies that pose the highest risk of adverse negative effects to all moving platforms driven on pavements and roads due to the diverse detrimental results that could occur, ranging from severe damage to the tyres, mechanical faults within the car, or total breakdown. The impact of any of those negative effects is grave when it comes to moving platform users, whether the moving platform is a road-based platform such as cars, lorries, etc. or a pavement-based platform such as a wheelchair, a mobility scooter, an electric scooter, or a bicycle. This could vary from discomfort to serious neck, or back injuries, and in some cases death. Not to forget the financial impact to users and local authorities when serious damage is inflicted onto the

moving platform as the users will need to repair their moving platform, and in the case of pothole damage to vehicles driven on the road, the local authority will be liable to pay compensation to the impacted persons. As there is not a single universal database to record injuries, falls, or damage caused by potholes, the only source of statistics is Freedom of Information (FOI) requests made by different sources to local authorities around England. Local authorities keep records of the number of potholes in their areas, the number of claims made, and compensations paid to members of the public for injuries, or damage resulted from potholes. In a recent study made by RAC [133], one of the major roadside assistance providers in the United Kingdom, there have been a recorded increase of 46% (+ 1893 cases) in car breakdowns caused by potholes in a period of one year, between the first quarter of 2022 and the first quarter of 2023. According to RAC, a recent FOI request made to 185 district and county councils, to which only 81 councils replied, shows that 556,658 potholes were reported in England during the period between Q1 2021, and Q1 2022. Another FOI request made by the Liberal Democrats [134] shows that a total of £1.77 million was paid in compensation to motorists between 2022 and 2023 by 85 local authorities who responded to the request. Claim numbers rose to 23,042 in 2022/23, nearly double the 13,579 claims made in 2021/22. As mentioned before, the data discussed only reflects the reported incidents recorded by only 85 out of 185 local authorities.

Whilst a large vehicle such as a car is more likely not to be severely damaged when crossing a pothole, assistive platforms such as EPWs and mobility scooters do not share the same likelihood. Users who rely on those platforms are at a higher risk of injury and in certain cases death when driving through a pothole, which means that assistive robotics cannot be considered safe for consumer use unless this issue is tackled. Due to the stochasticity in the occurrence of potholes, their non-linear shapes, sizes, depth, and texture, conventional detection techniques cannot be used and relied upon to safely navigate a platform. This brings the need for a more modern and novel technology that can bring forward a solution to this issue. As discussed in the previous chapter, and after discussing the different technologies which were used to detect potholes, it was concluded that the most promising solution is the one that involves deep learning. A solution to this

problem will form a significant leap towards a fully autonomous, or semi-autonomous (common control).

This chapter presents a multilayered computer vision system that uses machine vision fused with deep learning technology, more specifically, a convolutional neural network, and spatial understanding to detect and localise potholes in real-time scenarios. The first layer of the system consists of a trained neural network that detects potholes in real-time with the use of an RGB feed acquired via the an RGB-D camera. The network was trained on a manually collected dataset described in a previous chapter. The network is capable of achieving very high confidence rates with very high frame rates. The second layer of the system consists of a novel spatial measurement approach that uses depth imaging acquired via the depth feed of the RGB-D Camera. By using the depth feed, the system extracts the 3D real-world coordinates of certain points of interests from the objects detected within the first layer of the system and labelled as potholes. After aligning both frames together, the system will provide the user with accurate early warnings if the platform is within a distance that is considered risky to the safety of the user, providing the user, and the moving platform with the ability to safely navigate away from the pothole. This system can be trained to detect any negative object and will extract all the necessary information so that the negative object can be treated as a positive object when it comes to avoidance, whether it is within an autonomously driven system, or in a semi-autonomous one.

This chapter is organised as follows: statement of problem, system architecture, the training process are described in section 4.2. The results are discussed in section 4.3. The system's limitations are discussed in section 4.4, A live experiment of the system is illustrated in section 4.5, and the chapter's conclusion, and future improvements are presented in section 4.6

4.2 Methodology

4.2.1 Statement of Problem

Potholes are a challenge in their nature due to many different factors. Defined as depression and degradation within the road, or pavement surface, potholes are naturally stochastic and do not follow any defined pattern. They could have different shapes, depths, locations, orientations, and textures. Some potholes might be filled with rabble, leaves, dirt, and virtually anything else. This poses a severe challenge when it comes to detection rendering a significant number of the general detection techniques inadequate.

During the training process, the network would require a large input database to be used during the training and validation process. Due to the fact that potholes are serendipitous in nature and existence, collecting enough images to describe every instance of potholes existence is a very challenging task and requires images to be taken from different angles, distances, light and weather conditions, and potholes instances will need to vary from empty potholes to rabble-filled potholes, to water, or ice-filled potholed etc.

Most of the datasets currently available lack the universal coverage of the majority of the different stochastic appearances of potholes. Hence, the manually-collected dataset described in the previous chapter was assimilated.

The range of the detection also plays a very important role in assessing whether the system is usable or not. As the system can be mounted onto assistive vehicles such as mobility scooters or Electric-Powered Wheelchairs (EPWs), the system must detect objects at a high accuracy and within an adequate range to allow safe avoidance of the obstacle. The proposed system, as mentioned before, relies on an RGB-D camera instead of other conventional sensing techniques, the average range of RGB-D cameras is usually 5 to 10 meters when using professional, mid-range RGB-D cameras such as Intel RealSense D435i, Occipital Structure Core, or Zivid M400.

Not to forget the inevitable trade-off between speed and accuracy of the detection. As the system is to be mounted on autonomous or semi-autonomous platforms, real-time detection is mandatory, so the frame rate is pivotal when it comes to considerations to be taken when designing the system. To address this issue, a rigorous comparative

analysis of neural network performance will be undertaken. The primary selection criteria will prioritise models demonstrating exceptional accuracy and reliability, along with minimal computational overhead and complexity. Candidate architectures include DenseNet201, ResNet-50, and TinyYolo V4. Each network will undergo extensive training and evaluation to determine the optimal solution, balancing superior performance with efficient runtime characteristics.

4.2.2 System Architecture

Built upon a two-pronged architecture, the proposed system is represented in the figure below:

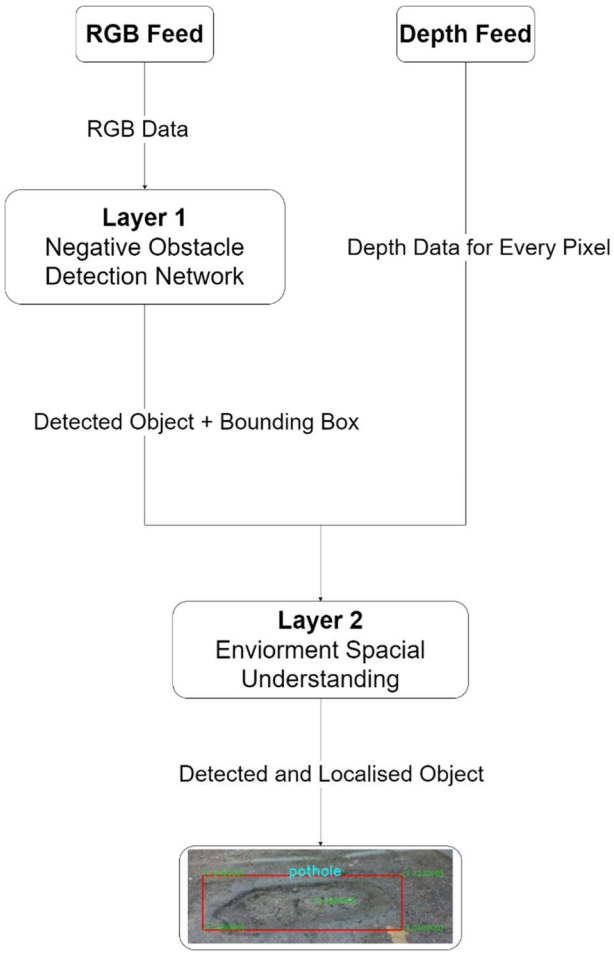


Figure 4.1: **Top-Level System Architecture**

The first layer of the system consists of a deep learning neural network whose input is an RGB continuous video feed provided via an RGB-D camera, a special RGB camera that provides an additional dedicated depth data feed. The camera can be mounted onto any platform regardless of its nature, whether it is a manually controlled vehicle, an Electric-powered wheelchair, a mobility scooter, an electric scooter, or a fully autonomous robot.

The acquired RGB data is processed by the core controller of the system and is fed into the real-time object detection network that identifies the candidate objects of interests and localises them within the 2D RGB frame. For every confirmed candidate where the detection confidence level is higher than 60%, four points of interests are identified and their X, and Y coordinates are recorded. Those four points are denoted by a, b, c, and d. These points represent the four corners of the rectangular bounding box that surrounds every successful detection. The 2D coordinates of the four points mentioned and the confidence level are recorded and passed onto the second layer for further processing and analysis. At this stage, the detection confidence, and speed should both be as high as possible, with a strong emphasis on the framerate or detection speed as the system is expected to run in real-time. For this, the network architecture chosen to perform the detection task is expected to be relatively small without compromising on the detection accuracy.

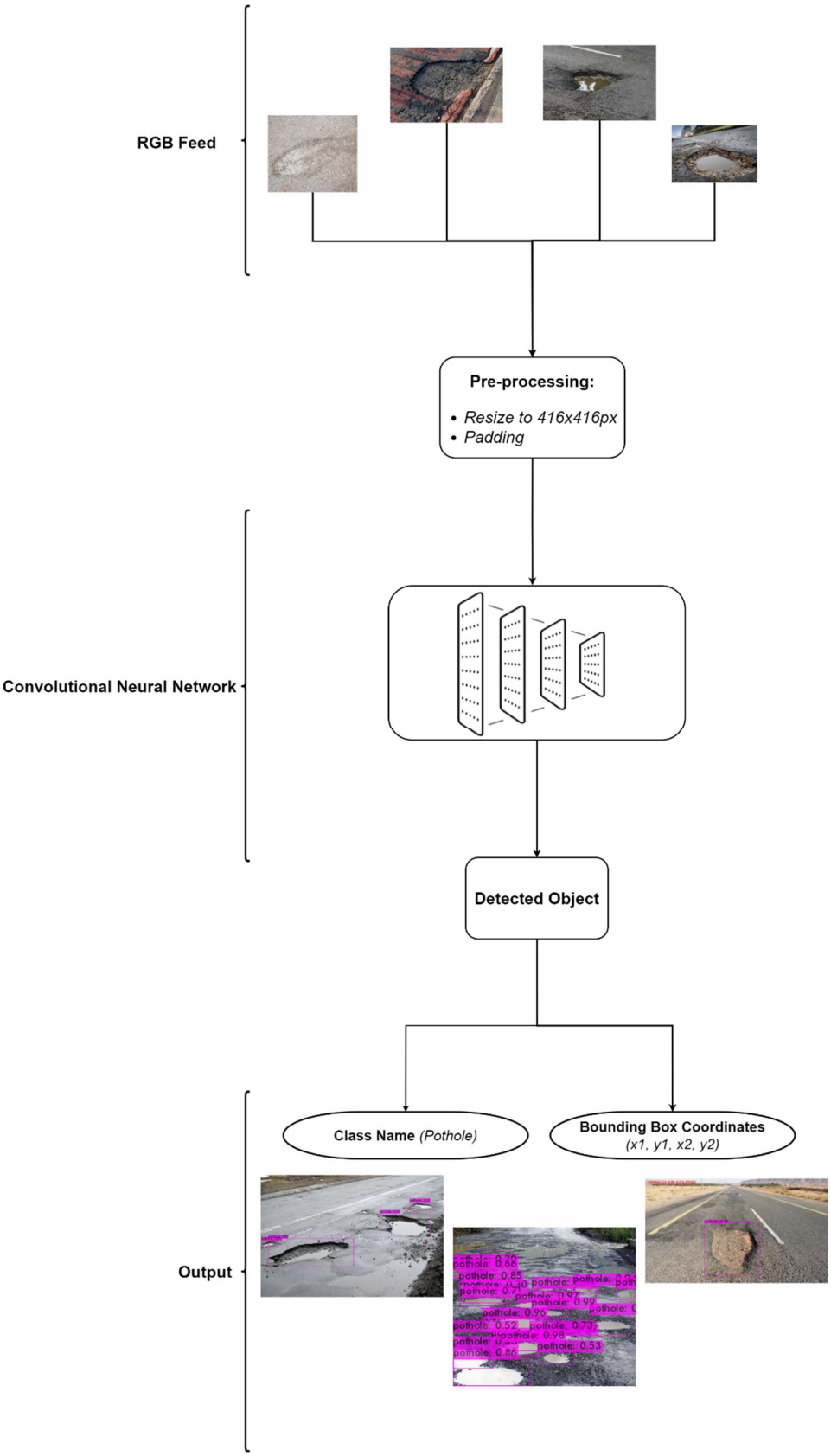


Figure 4.2: **Diagram representing the first layer of the system**

The second layer of the system comprises a novel spatial measurement approach that combines both the first layer, and the depth feed provided by the RGB-D camera after aligning both of them together in order to provide the necessary information to enable the negative obstacle to be treated as a positive one when applying any obstacle avoidance algorithm when it comes to autonomous driving, or to provide sufficient information to enable the user to safely avoid the negative obstacle in the case of semi-autonomous or manual systems. Aligning both frames essentially achieves a 3D projection from 2D coordinates to the 3D plane so that the system can provide real-life spatial measurements of the distance at which the four points of interest are located. The distance of every point from the centre of the camera is measured after acquiring the real-world X,Y and Z 3D-coordinates of every point within the four points of interest. A safety distance threshold of 1 meter have been chosen in order to provide sufficient time for the obstacle to be avoided no matter what driving mode is employed (manual, semi-autonomous, or full-autonomous).

The next sections cover more information about the neural network choice, training, and validation, as well as more detailed explanation on how the detection and localisation processes are achieved.

4.2.2.1 The Neural Network

Due to the complexity of the negative obstacle detection task, and the need for a high accuracy, and a very fast detection rate, three main object detection networks have been tested. The first network chosen was DenseNet-201 [135], the powerhouse image processing model widely known for its meticulously connected 201 layers whereby every layer receives specific pre-knowledge from the layer that precedes it. This empowers efficiency by reusing previous information about the

extracted features from different steps within the detection process. This network is renowned for achieving high accuracy in object detection and classification, it also enables the understanding of every single pixel's classification based on the objects that are detected. However, this network.

The next network that was used was ResNet50 [136], an object detection network famous for allowing the flow of data in two different directions, up and down, and allowing "connection skipping" whereby some layers are bypassed to avoid detection rate losses and gradients degradation. The network is formed of 50 residual blocks that, with the help of connection skipping enables the network to accurately pinpoint the objects of interest within an image with a very high precision. Every residual block as represented in Figure 4.3 is formed of three consecutive convolutional layers, the first one compresses the input channels in order to reduce the computational needs of the network, the next one performs feature extraction, focusing on spatial relationships and patterns within the image, and the third one decompresses the channels back to their original size. Then, identity mapping, or connection skipping takes place by adding the original input to the output of the last convolutional layer.

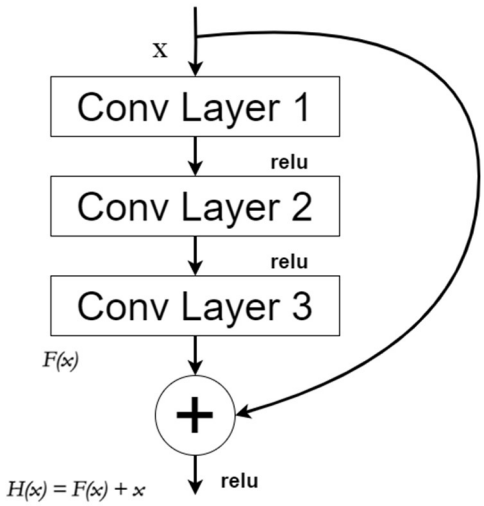


Figure 4.3: **ResNet residual block**

The third network chosen to be trained to perform the object detection task is the infamous You Only Look Once (YOLO) v4 [137]. YOLO is known for its fair blend of accuracy and speed. This network, unlike the other two networks used, combines the detection and classification processes into one making it a leading solution when it comes to real-time scenarios. It is composed of three main layers, the Backbone that performs feature extraction from the images via the "CSPDarkNet53", an innovative architecture that relies on residual connections along with a Cross Stage Partial (CSP) connections that enhances the flow of information and reduces complexity. The second layer of the network is the Neck, and it is where feature maps are refined and combined via two processes, the Spatial Pyramid Pooling (SPP) where information from different stages is aggregated, and a Path Aggregation Network (PAN) that combines high-level information or low-level ones. This means that it combines the semantic and the spatial features obtained. The last layer of the network is the head, where the prediction and classification is achieved, and where the bounding box, and confidence stores are determined. YOLO has many significant innovations that improve the gradient flow, accuracy, bounding box localisation. It also reduces computation cost and runtime and increases the training variability via data augmentation.

Due to the importance of the real-time detection and the need for a high framerate to be achieved, one of YOLO's compressed version Tiny YOLO, was chosen. Tiny YOLO's structure is simpler with reduced parameters. The network only has two YOLO heads at the detector level unlike its full version that has three, it also encompasses 29 pre-trained convolutional layers as opposed to 137 pre-trained convolutional layers as it is the case with the full version. However, Tiny YOLO can achieve approximately an 8-times faster framerate than YOLO v4, and when trained to detect a small number of classes, it is known to perform as good as its full version but with a very high frame rate enabling its use in real-time scenarios.

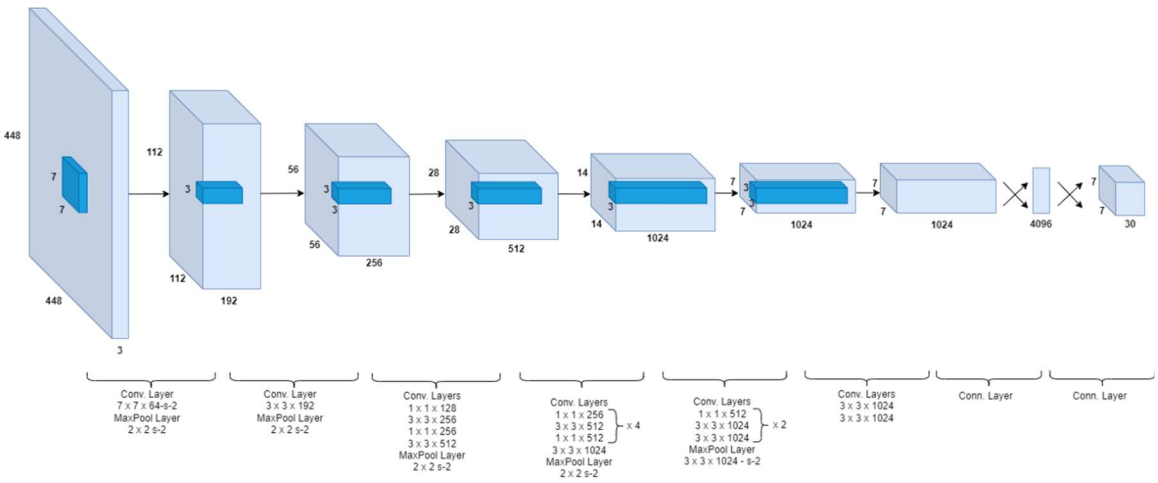


Figure 4.4: **Tiny YOLO Network Architecture**

The three chosen networks have been trained by using the manually collected dataset described in the previous chapter as the input dataset. This experiment was realised by using an AMD Ryzen 9 desktop computer with an 8-core/16-thread processor and an NVIDIA RTX3080 graphical processing unit (GPU) with 9GBs of Memory. The computer uses Linux as operating system and Darknet [137] as an open-source neural network framework.

4.2.2.2 Spatial Understanding of the Environment

Spatial understanding is achieved through the use of an RGB-D camera that provides, in addition to the normal RGB parameters, a layer of information representing the distance at which every pixel in the image is located with respect to the camera's sensor. This enables the system to have a clear understanding of its surroundings by mapping individual pixels to their respective distances from the camera. For this, the Intel D435i RGB-D camera was chosen as it offers accurate depth perception with a range of up to 10 meters. It utilises a global shutter sensor that enables the camera to produce high quality video frames even in low-light environments.

The first step consists of aligning the Depth feed with the RGB feed. This can be achieved via intel's own per-pixel geometric transformation algorithm via intrinsic and extrinsic calibrations. The intrinsic calibration ensures that the RGB and Depth sensors are calibrated to adhere to distortions and to improve accuracy, the extrinsic calibration determines the accurate spatial relationship between both sensors (RGB and Depth), this is achieved by finding the rotation and translation needed to align both feeds together. As a result, the 2D (X,Y) coordinates are projected onto spatial 3D (X,Y,Z) coordinates within the real-world. This means that the distance at which every pixel is located with respect to the camera sensor is calculated.

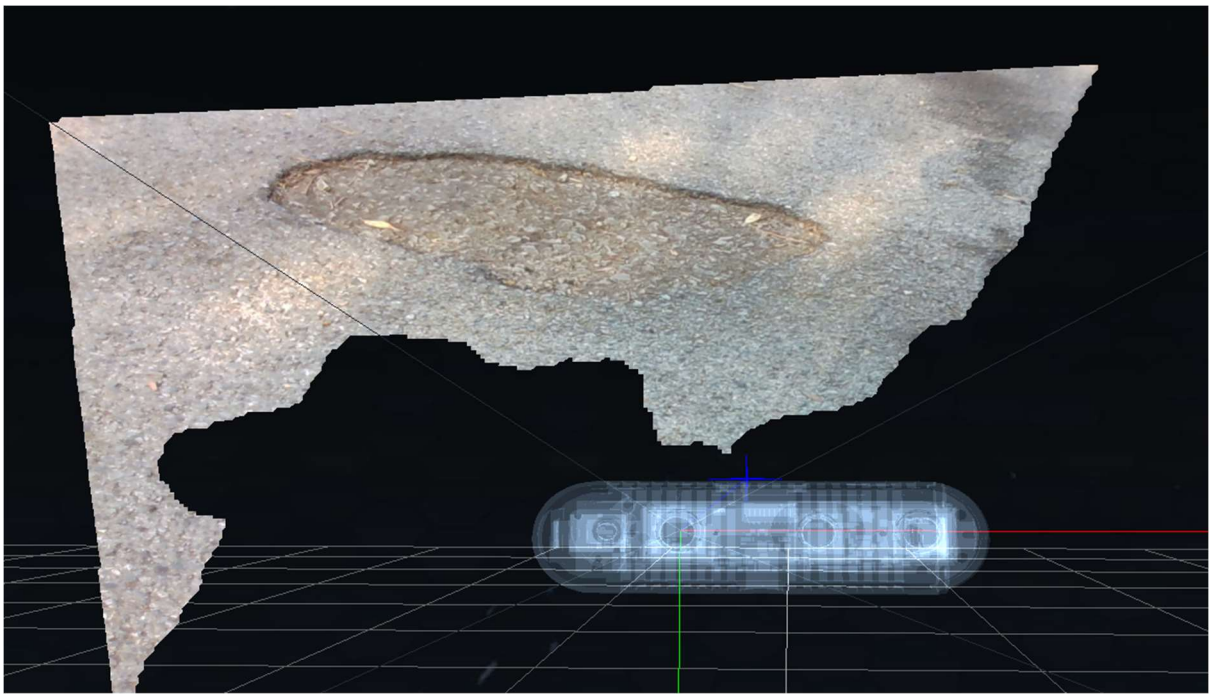


Figure 4.5: **Pothole as seen by the RGB-D Camera**. This represents the resultant of the alignment of the RGB and the Depth frames together. The Intel RealSense D435i camera's sensor is capable of measuring the distance between its sensor and any pixel within the detected frame.

In order to provide an accurate measurement, the chosen camera's specifications have been taken into account. It has been concluded that the camera's design and build do not affect the measurements and no model-specific adjustments are needed based on the below:

The origin of the camera is 17.5mm from the centreline of the left imaging sensor which as per Figure 4.9, this distance is a negligible value.

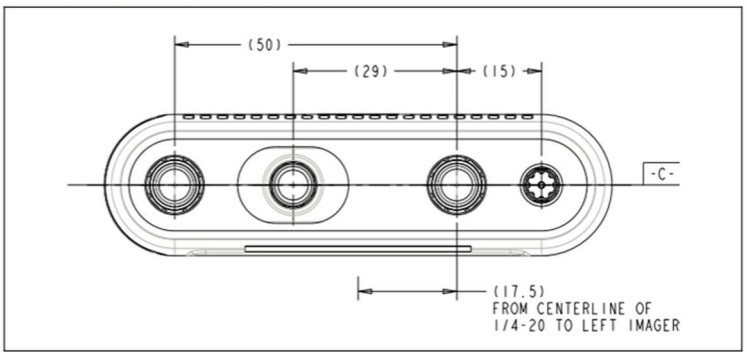


Figure 4.6: **Location of the imaging sensor within the Intel D435i Camera** [138]

The measured depth of the point of interest is obtained via the depth frame. This process involves a fusion between two methods, stereo vision, and an infrared sensor (IR) in order to improve accuracy. RealSense D435i's depth measurement start point is the surface of its sensor covered by a protective covering glass with a thickness of 4.2mm. As this is a negligible value, the depth calculation starting point is considered the surface of the glass cover of the camera.

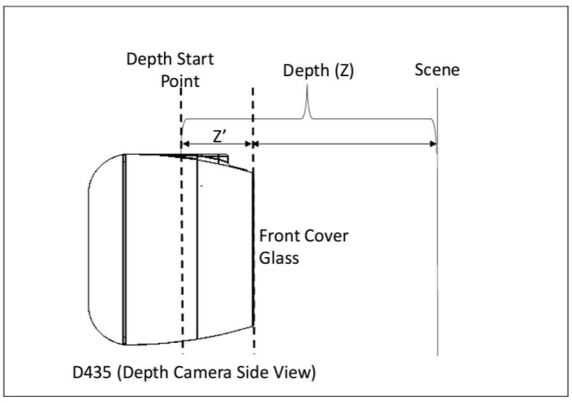


Figure 4.7: **Depth calculation start point** [138]

With the camera's specifications being taken into consideration, the system's spatial understanding is achieved by first measuring the distance between the camera's origin and each of the corners of the bounding box starting from the upper-left side of the observed and detected bounding box, denoted by A in figure 4.11, the next point of interest is the lower-left corner denoted by B, followed by the upper-right corner denoted by C, and the lower-right corner denoted by D. The distance between the camera and the centre of the detected object is also measured and denoted by E in order to provide additional knowledge and potentially estimate the depth of the negative object if needed. It is

important to note that the corner points of interests are named according to the point of view (POV) of the camera with respect to the detected objects.

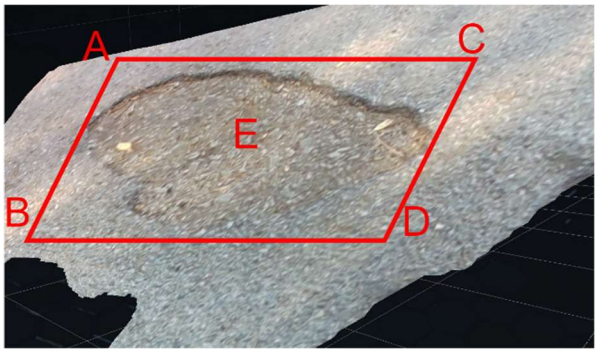


Figure 4.8: **The four main points of interest (POIs), in addition to the centre point.** These points are denoted by A, B, C, D, and the centre point E named based on the point of view of the camera.

Irrespective of the camera's POV, and the moving platform's location, only three POIs will need to be considered to achieve safe navigation and avoidance. Based on Figure 4.11's POV, the main points to be considered are B, D, and C as B, and D are the closest points to the moving platform and if the user keeps a safe distance from those points, the user is safe. If the user or the autonomous driving algorithm decides to go right, which is the driving orientation that is taken upon observing the pothole represented in the figure, point C will be of interest along with D as the user or algorithm will need to keep a safe distance from both points. For added safety and security and to minimise the risk of errors, 1 meter is considered to be the safe distance threshold, i.e. the user or algorithm are prompted to begin the safe avoidance process as soon as the distance between the camera and any of the three POIs is less than or equal to 1 meter. This novel avoidance approach will ensure that the moving platform, regardless of its type, will consistently be kept at a safe distance from the negative obstacle. The system will also provide the user or the navigating algorithm with enough information to treat the negative obstacle the same way a positive obstacle is treated without the need to significant changes to be added to the autonomous algorithm, or to the semi-autonomous, or manual navigation systems.

4.3 Experiment

A real-life experimental apparatus has been set up in order to evaluate the system's performance in different conditions and scenarios. For this, the Intel RealSense D435i camera has been mounted onto top the left foot holder of a Salsa M2 Electric-powered wheelchair at a height of 35cm from the ground as per Figure 4.12. The tilting angle of the camera is 35 degrees to ensure that the camera can have a clear POV covering the front side of the footrests up to a 4-meter distance. The EPW was driven around the University of Kent's Canterbury Campus during different periods of the day, and different weather conditions within the area represented in Figure 4.13.

Figure 4.9: **RGB-D Camera mounted onto the EPW.**

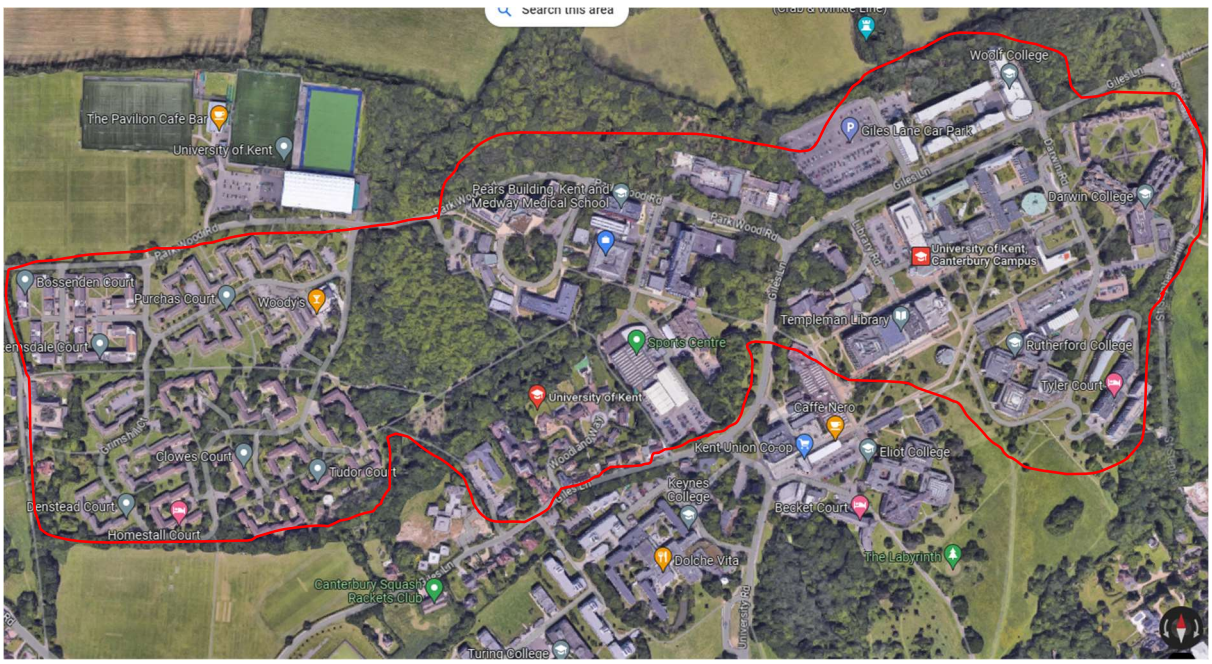


Figure 4.10: **Experiment location and boundary.** The University of Kent Canterbury Campus at which the experiment took place within the boundary marked in red in different periods of the day and in different weather conditions.

4.4 Results and Discussion

4.4.1 Network Training

In order to classify the detection as a True Positive, the detection will need its maximal overlap of the detected boundary box with the ground truth to be larger then or equal the Intersection over Union (IoU) ratio defined within the training process. The IoU is calculated via the equation below:

$$IoU\ (\alpha, \beta) = \frac{|\alpha \cap \beta|}{|\alpha \cup \beta|} \qquad (4.1)$$

Where α represents the detected bounding box, and β represents the ground truth. The next step is to calculate the precision for every detection in order to obtain a unified measure of precision performance assessment, the mean average precision (mAP). This involves outlining a precision envelope, which essentially captures the total area under the curve by focusing on specific points where recall values shift.

The precision, recall, and F-1 Score are calculated as follows:

$$Precision = \frac{TP}{TP+FP} \qquad (4.2)$$

$$Recall = \frac{TP}{TP + FN} \qquad (4.3)$$

$$F1 - Score = \frac{2\ \times\ Precision\ \times\ Recall}{Precision + Recall} \qquad (4.4)$$

Where TP is the number of correct detections, or true positives, FP is the number of incorrect detections, or false positives, and FN is the number of incorrect rejections, or false negatives.

The mean average precision is then obtained by summing the average precision (AP), or the area under the precision-recall curve obtained within the training process which obtained by calculating the average of all the Precision values, and diving AP by the total number of detections:

$$mAP = \frac{\sum_{i=1}^{N} AP}{N} \qquad (4.5)$$

To mitigate the risk of overfitting and encourage robust learning, the dataset was randomly split into training (80%) and validation (20%) sets. This stratified sampling approach ensures the model encounters diverse examples during both training and evaluation.

The parameters were unified for all three networks, and are as follows:

- IoU = 70 i.e. a detection is classified as a true positive if the overlap between the detected bounding box, and the ground truth is 70% or more.
- Learning Rate = 1e-4
- Training Images = 570 randomised (80% of the dataset)
- Validation Images = 143 randomised (20% of the dataset)

DenseNet201:

The network training took 27.5 hours and was completed after 45000 epochs with an average loss of 0.0227 , and an mAP of 98.3% as illustrated in Figure 4.11.

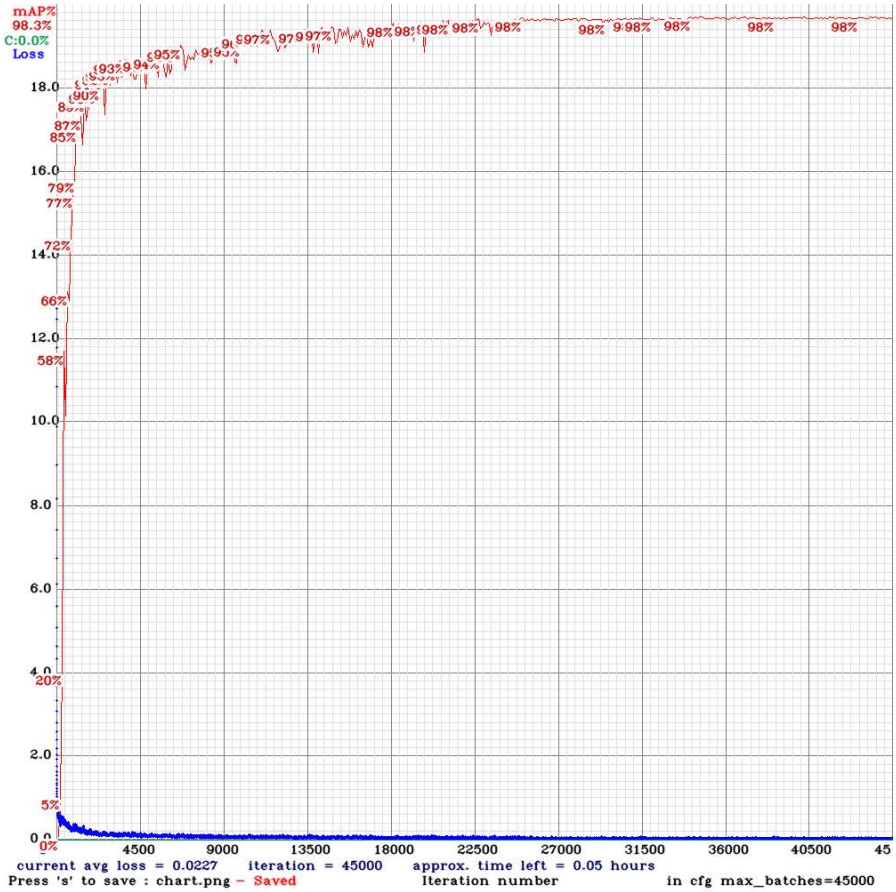


Figure 4.11: **DenseNet 201 training resulting graph**. The graph illustrates the valuation loss, and the mean average precision (mAP) across 45000 iterations. The blue line represents the valuation loss, and the red line represents the mean average precision

Resnet50:

The training ended after 21.7 hours at 23000 epochs with an average loss of 0.0391, and a mean average precision mAP of 95.9% as illustrated in Figure 4.12.

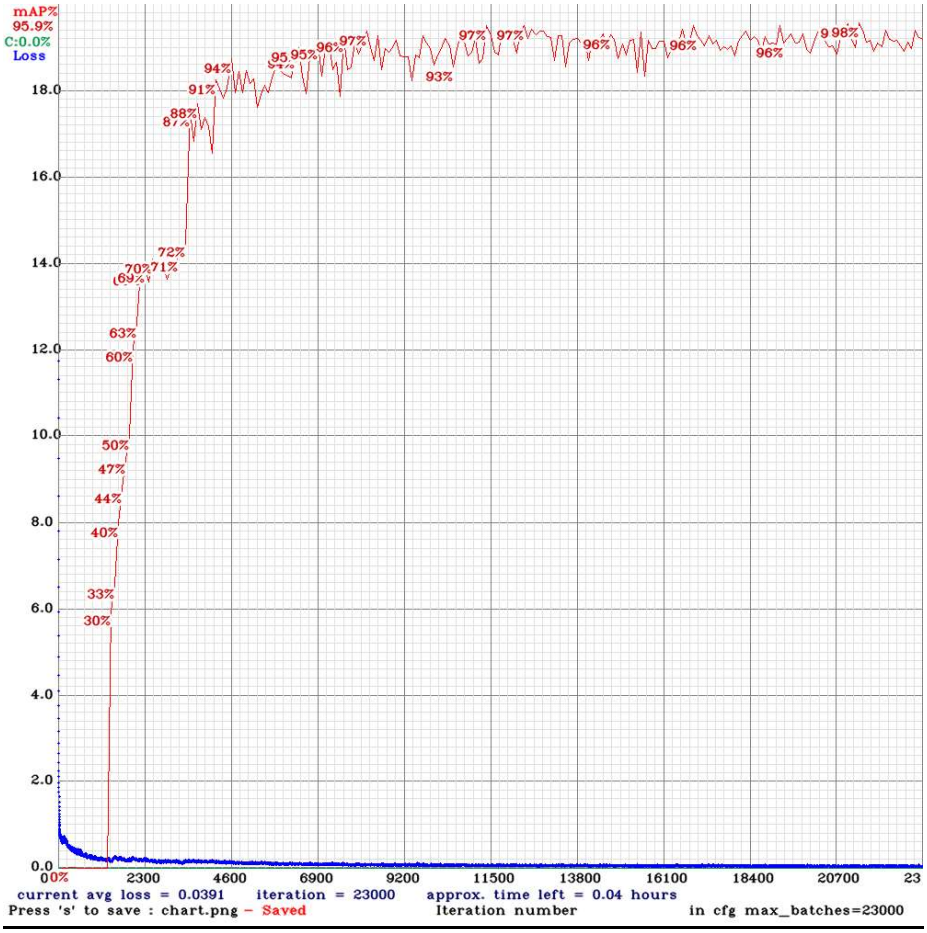


Figure 4.12: **ResNet50 training resulting graph.** The graph illustrates the valuation loss, and the mean average precision (mAP) across 23000 iterations. The blue line represents the valuation loss, and the red line represents the mean average precision

<u>Tiny YOLO:</u>

The training ended at 2000 epochs after 8.7 with an average loss of 0.3332, and a mean average precision mAP of 97.7% as illustrated in Figure 4.13.

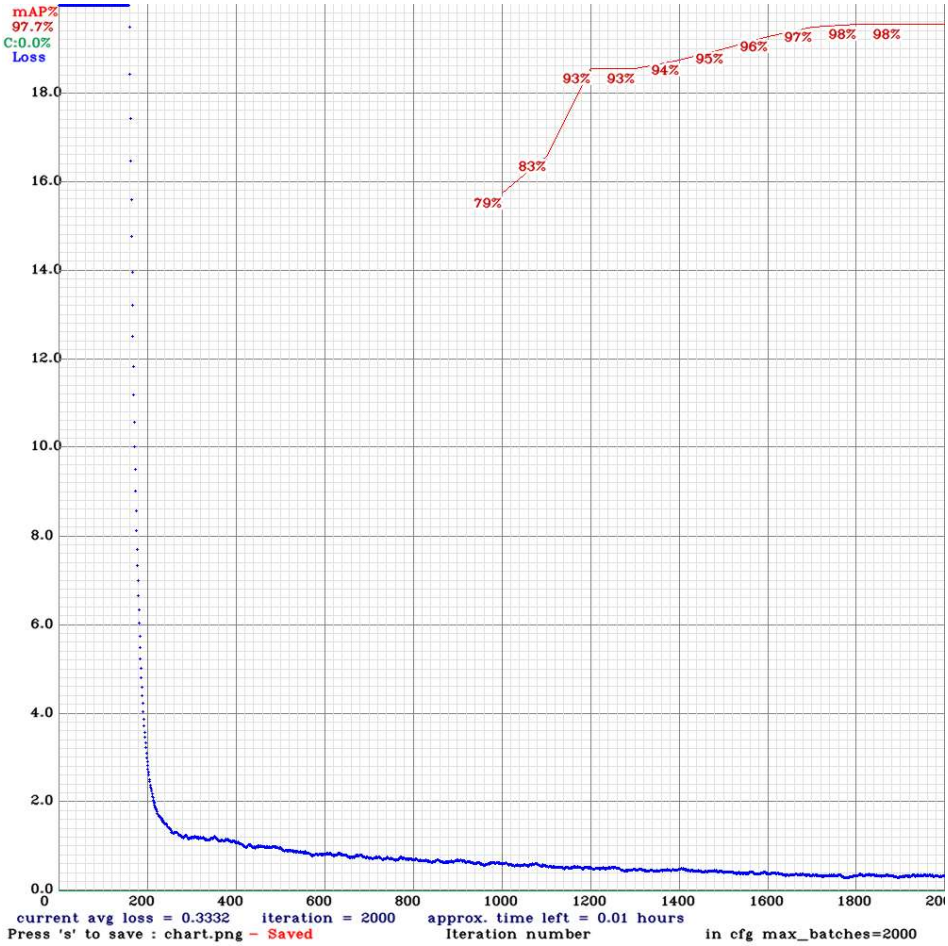


Figure 4.13: **Tiny YOLO training resulting graph**. The graph illustrates the valuation loss, and the mean average precision (mAP) across 2000 iterations. The blue line represents the valuation loss, and the red line represents the mean average precision.

The performance metrics of the networks are illustrated below:

Table 4.1: **Performance metrics for all the custom-trained networks.**

Metric / Network	AP	TP	FP	FN	Precision	Recall	F1-Score
Densenet 201	98.34%	1116	64	23	0.945	0.979	0.961
Resnet 50	95.90%	1117	112	22	0.908	0.980	0.943
Tiny YOLO	**98.85%**	1118	30	20	**0.964**	**0.98**	**0.982**

Several different inputs were given to all three networks in order to assess its performance, samples of those results are represented within Table 4.2.

Table 4.2: **Detection accuracy comparison table for all three networks.** This table illustrates the performance of all three networks, DenseNet201, ResNet50, and Tiny YOLO v4, when applied to four different photos. The first row represents the performance on a clear reflective water-filled pothole; the second row represents the performance of all three networks when applied to a photo showing two semi-reflective water-filled potholes. The third row represents the performance of the networks when applied to a photo showing one rabble-filled pothole, and the last row represents a challenging scenario, numerous reflective water-filled potholes with a contour that is very difficult to see and identify.

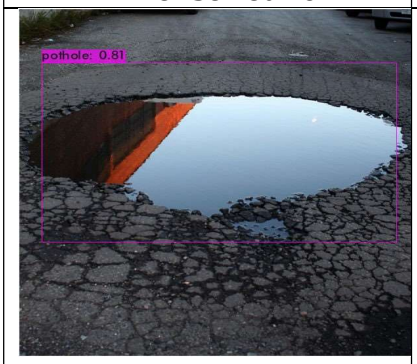
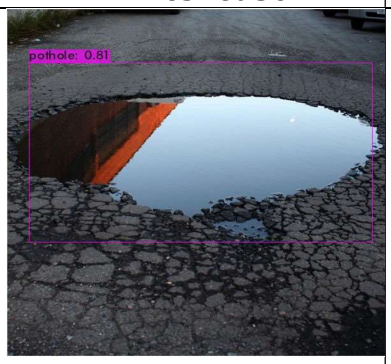
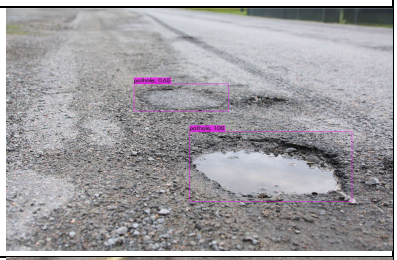
DenseNet 201	ResNet 50	Tiny YOLO V4
		
		

Table 4.1 illustrates the three chosen networks DenseNet 201, ResNet 50, and TinyYOLO v4 in four different scenarios. The first row is a clear reflective water-filled pothole, the second row represents two semi-reflective water filled potholes, whilst the third row represents one rabble-filled pothole. The last row in the table represents what is considered a challenging case whereby the scene contains numerous water-over-filled

reflective potholes with their contours being almost embedded with the road. It is obvious that TinyYOLO is the highest-achieving network with more objects being detected and high detection rates achieved with 96% confidence rate achieved on the first scenario, 100% and 68% confidence for the second scenario, 99% on the third, and 6 potholes detected within the fourth scenario with confidence rates ranging between 34% for one very challenging pothole and 95%. DenseNet 201 comes in second with 81% detection rate achieved within the first scenario, 94% and 69% in the second, 92% in the third scenario, , and only two detections within the fourth scenario with 94% and 33% as confidences. As for ResNet 50, it comes in third, with 81% confidence in the first scenario, no detection in the second and third scenarios, four detections in the fourth scenario with confidence rates ranging between 30% and 72%.

In order to better assess the networks, another testing attempt has been made. A a publically-available online video [6] has been applied onto DenseNet201, ResNet50, and Tiny YOLO v4. The video contained different instances of potholes. Three consecutive frames were extracted and used within the comparison process as represented in Figure 4.14. TinyYOLO v4 came in first with nine potholes detected in the first frame with confidences ranging between 34% and 97%, eight potholes detected in the second frame with confidences ranging between 60% and 100%, and eight potholes detected in the third frame with confidences ranging between 30% and 99%. DenseNet 201 came in second with five detections in the first frame, and confidences ranging between 27% and 90%, five detections in the second frame, and confidences ranging between 42% and 87%, and six detections within the third frame, and confidence levels ranging between 47%, and 87%. ResNet 50 came in last with four detections in the first frame having confidences ranging between 58% and 89%, four detections in the second frame with confidences ranging between 67% and 89%, and five detections in the third frame with confidences ranging between 43% and 85%.

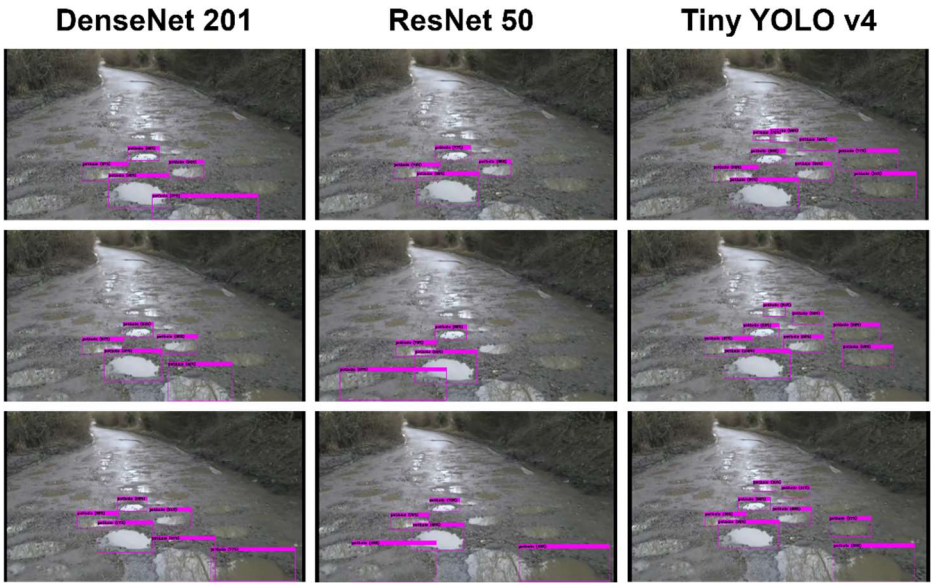


Figure 4.14: **Video comparison of the three networks.** Tests results on a publically-available online video [139] with three consecutive frames extracted. Tiny YOLO v4 clearly shows

After assessing DenseNet201, ResNet50, and TinyYOLO v4, it can be concluded that TinyYOLO v4 is a better and more robust and reliable detector for the autonomous, semi-autonomous, or manual avoidance of potholes in different scenarios in real-life situations. Tiny YOLO v4 was chosen as the detector for the first layer of the system. For additional reassurance, Tiny YOLO v4 was subjected to further testing with more challenging scenarios as represented in Figure 4.15

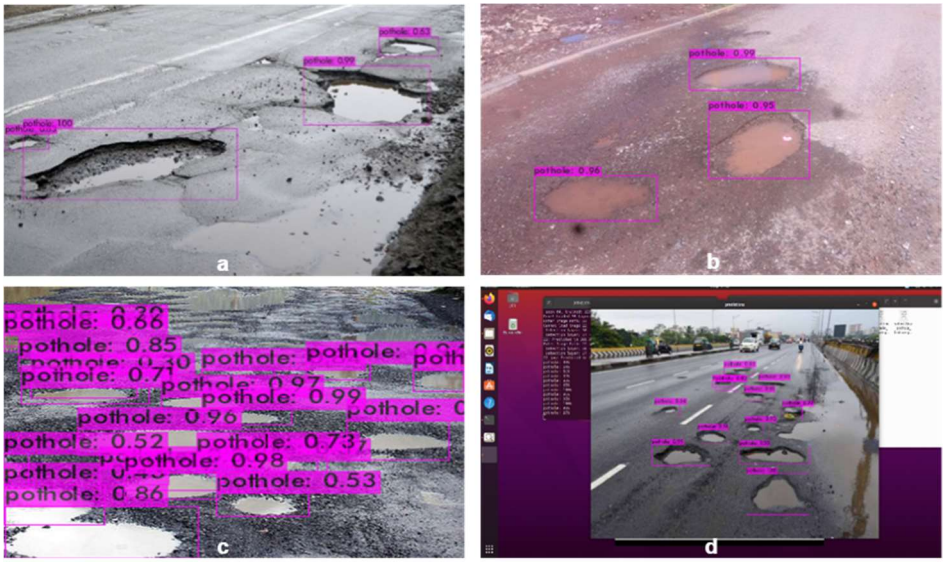


Figure 4.15: **Tiny YOLO v4 Confidence Rate**. Tests results when applied onto different cases of triple clear (a), and muddy (b) water-filled potholes, and on a large number of reflective water-filled (c) potholes, and non-reflective water-filled potholes (d).

Tiny YOLO's result evidently shows that the confidence achieved is very high as it can be seen from the Figure 4.9. The accuracy of the detection in the case of clear and reflective water-filled potholes (a), and muddy water-filled potholes (b), the confidence rate is higher than 94% in 2 out of 3 potholes in (a), and in all of three potholes in (b). As for the cases of multiple water-filled potholed (c), and (d), the results confidence rate is always higher than 60% except for some minor individual cases where the pothole is barely noticeable for the naked eye.

Assessing the speed of the detection is vital to ensure that the system can run in real-time, for this, the speed limit for on-road vehicles in residential areas in England (30 mph) has been considered within the assessment. This speed is equivalent to 48.28 km/h or 14.411 m/s. As for assistive technologies, the maximum speed of an electric-powered wheelchair is 4 mph, or 6 km/h, 1.67 m/s. In addition, upon testing the network with an RGB live video feed as an input, the processing rate in frames per second, or inference rate obtained is 1050 FPS.

The above values are incorporated within the detection rate equation:

$$Frame\ Processing\ Rate\ (frames\ per\ meter) = \frac{Frame\ Rate\ (fps)}{Max\ Speed\ (\frac{m}{s})}\ \text{fpm} \quad (4.6)$$

By applying the different speeds discussed earlier, along with the framerate obtained, the Frame Processing Rate (FPR) can be assessed as follows

$$Frame\ Processing\ Rate\ \ 1\ (FPR1) = \frac{1050}{14.411} = 78.29\ fpm\ for\ a\ vehicle\ at\ 30\ mph \tag{4.6}$$

$$Frame\ Processing\ Rate\ \ 2\ (FPR2) = \frac{1050}{4.3333} = 315.03\ fpm\ for\ an\ EPW\ at\ 8\ mph \tag{4.7}$$

$$Frame\ Processing\ Rate\ \ 3\ (FPR3) = \frac{1050}{1.97} = 532.99\ fpm\ for\ an\ EPW\ at\ 4\ mph \tag{4.8}$$

It is commonly known that a detection is considered real-time if the framerate of the detection is 30 fps. Tiny YOLO achieved a 30-times faster detection rate achieving a processing rate 78.29 frames every meter when mounted onto a vehicle moving at a speed of 30 mph, it has achieved 315.03 frames every meter when mounted onto an EPW, or a mobility scooter moving at 8 mph, and 532.99 frames every meter when the EPW or mobility scooter is cruising at 4 mph. This provides the user, or the avoidance algorithm sufficient time to perform the required avoidance manoeuvers. The achieved speeds confirm that Tiny YOLO is the optimal choice providing the efficacious equilibrium between high confidence, and high detection rate, or speed. This frame-rate along with the mentioned accuracy were obtained when Tiny YOLO was evaluated with the publically-available online video [6] published on YouTube by the BBC. Three additional frames other than the ones used in Figure 4.14 have been extracted and illustrated in Figure 4.16.

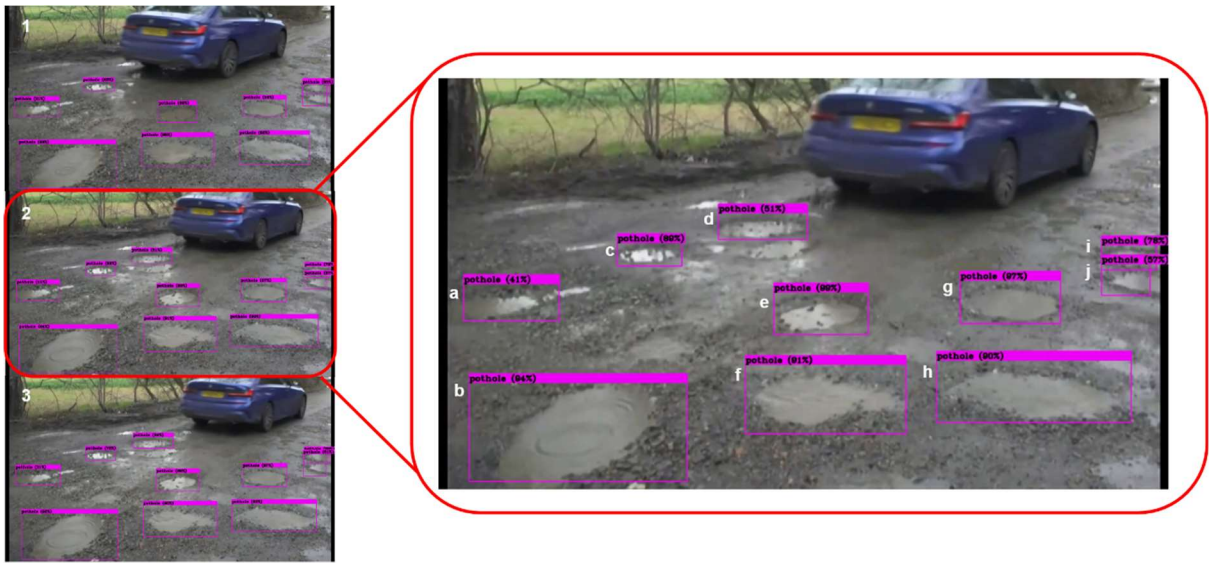


Figure 4.16: **Real-time testing of the YOLO Network.** Tiny YOLO with a publically-available online video [139] used as input, with three consecutive frames have been extracted. In the test above, the precision rate of the detection is respectively: pothole 1.a: 85%, pothole 1.b: 73%, and pothole 1.c: 95% pothole 2.a: 56%, pothole 2.b: 82%, and pothole 2.c: 97%. Pothole 3.a: 53%, pothole 3.b: 70%, and pothole 3.c: 96%. The Average Frame Rate achieved is 52 fps which is considered real-time.

Thus far, the system's detector, Tiny YOLO, chosen, is able to accurately detect potholes in real-time, and early enough to safely avoid them. The output of the network is the X, Y coordinates of every point within the four corners of the bounding box, in 2D form, along with the confidence percentage, or detection accuracy for every pothole object detected. However, there is no spatial knowledge of the surrounding environment, which means that the system lacks important information needed as part of the avoidance process. The next section discusses the second layer of the system, where the spatial understanding is performed, providing the system with the last crucial datum.

4.4.2 Spatial Understanding of the Environment

During the experiment that took place in the University of Kent, Canterbury Campus as described in Section 4.3 within the path illustrated in Figure 4.16, detected potholes were observed from three different positions. As a result, three points of interest were pinpointed and the distance between them and the centre of the camera measured by the system was recorded along with two additional points for an added assessment of the performance of the system. The results obtained by the system were recorded and labelled "Algorithm", or "ALG". The resulting dataset contains 105 different test cases.

A tape measure was used to measure the ground truth which was recorded as "Ground Truth" or "GT". The accuracy of the detection, or confidence returned by the system was also recorded.

A live demonstration of the system's performance in real-time is presented within the following video (video). This video exhibits the system's performance when the wheelchair is driven at its maximum speed (6 mph).

Another video (video) is provided to illustrate the system's performance in the case of water-filled potholes

The resultant dataset is illustrated within Table 4.2.

Table 4.4.3: **Experiment Results.** This table represents the results obtained within the experiment described in Section 4.3 in 105 different test cases. Each examined pothole is assessed from three different points of view representing three unique test cases.

Pothole #	Accuracy	Top Left – CAM	Top Left – REAL	Top Right – CAM	Top Right – REAL	Bottom Left – CAM	Bottom Left – REAL	Bottom Right – CAM	Bottom Right – REAL	Center – CAM	Center – REAL
1	99.29	**2.52**	2.38	**2.384**	2.46	**1.826**	1.84	**1.63**	1.68	**2.243**	2.3
1	94.53	**1.84**	1.9	**1.73**	1.72	**1.141**	1.17	**1.03**	1.03	**1.48**	1.4
1	89.78	**1.892**	1.8	**1.591**	1.6	**1.087**	1.1	**0.98**	1	**1.32**	1.3

2	75.23	**1.673**	1.64	**1.652**	1.66	**1.124**	1.131	**0.979**	1.13	**1.405**	1.4
2	74.56	**2.86**	2.1	**1.968**	1.93	**1.19**	1.25	**1.159**	1.17	**1.575**	1.61
2	81.363	**1.98**	1.91	**1.879**	1.82	**1.284**	1.31	**1.23**	1.27	**1.646**	1.6
3	83.753	**2.929**	2.78	**2.677**	2.64	**1.955**	1.94	**1.851**	1.84	**2.39**	2.32
3	91.76	**2.418**	2.36	**1.962**	1.93	**1.399**	1.41	**1.19**	1.24	**1.782**	1.74
3	73.9	**1.529**	1.51	**1.434**	1.42	**0**	0.87	**0**	0.82	**1.202**	1.21
4	96.138	**3.336**	3.21	**2.797**	2.73	**2.032**	2.03	**1.808**	1.9	**2.495**	2.41
4	98.84	**2.019**	1.97	**2.025**	2.06	**1.368**	1.34	**1.356**	1.34	**1.655**	1.65
4	93.758	**2.327**	2.3	**2.1**	2.12	**1.35**	1.37	**1.245**	1.25	**1.757**	1.77
5	90.958	**2.136**	2.1	**2.345**	2.2	**1.51**	1.56	**1.587**	1.58	**1.857**	1.8
5	94.556	**1.591**	1.56	**1.629**	1.61	**1.202**	1.19	**1.211**	1.26	**1.441**	1.41
5	96.124	**1.454**	1.43	**1.468**	1.43	**1.033**	1.04	**1.052**	1.05	**1.214**	1.22
6	99.25	**1.587**	1.55	**1.633**	1.61	**1.238**	1.25	**1.268**	1.26	**1.478**	1.455
6	97.249	**1.718**	1.71	**1.767**	1.72	**1.04**	1.05	**1.066**	1.07	**1.368**	1.36
6	98.8	**1.5**	1.47	**1.548**	1.52	**0.981**	0.9	**0.975**	0.97	**1.223**	1.26
7	70.39	**1.742**	1.72	**1.777**	1.77	**0.996**	0.99	**0.998**	0.99	**1.322**	1.32
7	76.449	**1.798**	1.79	**2.039**	2.02	**1.177**	1.17	**1.258**	1.24	**1.51**	1.51
7	69.56	**1.612**	1.56	**1.704**	1.7	**1.098**	1.1	**1.116**	1.117	**1.368**	1.34
8	86.317	**1.84**	1.79	**2.019**	1.99	**1.3**	1.32	**1.431**	1.42	**1.591**	1.59
8	96.357	**1.914**	1.87	**1.902**	1.89	**1.281**	1.271	**1.289**	1.289	**1.642**	1.62
8	79.231	**1.612**	1.59	**1.563**	1.53	**1.098**	1.1	**1.043**	1.05	**1.302**	1.3
9	96.492	**1.482**	1.44	**1.521**	1.49	**1.106**	1.1	**1.14**	1.16	**1.327**	1.32
9	95.017	**1.955**	1.9	**1.782**	1.75	**1.431**	1.42	**1.347**	1.35	**1.629**	1.61
9	97.689	**1.485**	1.45	**1.408**	1.41	**1.235**	1.22	**1.118**	1.118	**1.322**	1.32
10	66.37	**1.71**	1.71	**1.591**	1.6	**1.313**	1.3	**1.235**	1.26	**1.548**	1.52
10	86.237	**1.485**	1.46	**1.56**	1.52	**1.057**	1.05	**1.116**	1.113	**1.305**	1.3
10	89.207	**1.457**	1.42	**1.536**	1.53	**1.102**	1.11	**1.136**	1.13	**1.292**	1.29
11	99.1	**1.819**	1.81	**1.824**	1.82	**1.233**	1.23	**1.228**	1.22	**1.507**	1.5
11	96.949	**2.122**	2.1	**1.949**	1.94	**1.302**	1.3	**1.223**	1.23	**1.1563**	1.55
11	99.223	**2.1**	2	**1.691**	1.69	**1.253**	1.25	**1.145**	1.14	**1.478**	1.48
12	95.555	**1.808**	1.78	**1.681**	1.67	**1.132**	1.28	**1.068**	1.07	**1.5**	1.5
12	94.8609	**1.993**	1.99	**1.862**	1.86	**1.308**	1.3	**1.104**	1.12	**1.587**	1.57
12	90.3716	**2.144**	2.1	**1.824**	1.83	**1.629**	1.61	**1.336**	1.33	**1.868**	1.85
13	97.118	**1.793**	1.75	**1.808**	1.79	**1.153**	1.15	**1.136**	1.13	**1.421**	1.41
13	93.9	**2.985**	2.94	**2.942**	2.93	**2.293**	2.29	**2.229**	2.223	**2.66**	2.254

13	95.624	**3.354**	3.39	**3.088**	3.1	**2.557**	2.55	**2.336**	2.3	**2.561**	2.52
14	93.722	**3.135**	2.96	**3.007**	2.94	**2.427**	2.4	**2.456**	2.4	**2.415**	2.6
14	95.133	**1.945**	1.92	**1.987**	1.97	**1.447**	1.44	**1.444**	1.45	**1.686**	1.68
14	96.142	**1.885**	1.82	**1.777**	1.77	**1.35**	1.35	**1.294**	1.31	**1.552**	1.54
15	93.163	**1.587**	1.57	**1.583**	1.58	**1.38**	1.38	**1.431**	1.43	**1.5**	1.5
15	94.749	**1.51**	1.51	**1.503**	1.5	**1.248**	1.25	**1.209**	1.21	**1.405**	1.4
15	92.987	**1.868**	1.84	**1.925**	1.87	**1.579**	1.58	**1.62**	1.63	**1.764**	1.77
16	91.96	**1.974**	1.95	**2.045**	1.97	**1.7**	1.67	**1.646**	1.65	**1.882**	1.84
16	92.85	**1.962**	1.93	**1.931**	1.92	**1.563**	1.56	**1.567**	1.56	**1.662**	1.7
16	96.183	**2.174**	2.05	**2.244**	2.1	**1.846**	1.8	**1.902**	1.899	**1.965**	1.97
17	74.224	**3.199**	3	**3.318**	3.1	**2.418**	2.4	**2.425**	2.43	**2.524**	2.51
17	85.286	**2.39**	2.35	**2.336**	2.3	**1.742**	1.74	**1.695**	1.7	**1.992**	2
17	78.214	**2.252**	2.25	**2.42**	2.39	**1.51**	1.52	**1.63**	1.64	**1.902**	1.89
18	81.122	**2.039**	1.99	**2.399**	2.2	**1.347**	1.35	**1.461**	1.46	**1.782**	1.78
18	92.3	**2.174**	2.1	**1.782**	1.77	**1.431**	1.43	**1.281**	1.28	**1.718**	1.67
18	92.34	**2.22**	2.15	**1.955**	1.94	**1.374**	1.37	**1.276**	1.27	**1.681**	1.68
19	71.6822	**1.943**	1.9	**1.879**	1.86	**1.386**	1.38	**1.402**	1.4	**1.723**	1.71
19	69.8	**1.782**	1.77	**1.908**	1.9	**1.308**	1.3	**1.35**	1.35	**1.699**	1.67
19	67.5	**1.968**	1.9	**1.595**	1.58	**1.019**	1	**0.857**	0.96	**1.258**	1.24
20	84.19	**1.937**	1.89	**1.851**	1.84	**1.444**	1.44	**1.402**	1.4	**1.659**	1.63
20	67.6	**1.955**	1.94	**2.032**	2	**1.365**	1.36	**1.353**	1.35	**1.7**	1.66
20	94.09	**1.772**	1.76	**1.808**	1.8	**1.162**	1.16	**1.083**	1.1	**1.399**	1.4
21	98.281	**1.604**	1.59	**1.604**	1.59	**1.245**	1.24	**1.218**	1.21	**1.461**	1.46
21	92.55	**1.798**	1.79	**1.84**	1.82	**1.437**	1.43	**1.399**	1.4	**1672**	1.66
21	91.419	**1.686**	1.67	**1.668**	1.68	**1.389**	1.38	**1.204**	1.2	**1.51**	1.5
22	95.325	**2.252**	2.22	**2.31**	2.28	**1.762**	1.76	**1.728**	1.72	**1.832**	1.88
22	91.09	**1.999**	1.94	**1.925**	1.9	**1.646**	1.64	**1.552**	1.5	**1.722**	1.72
22	89.722	**1.444**	1.47	**1.389**	1.41	**1.207**	1.2	**1.193**	1.19	**1.26**	1.28
23	83.667	**2.032**	1.97	**1.908**	1.88	**1.514**	1.51	**1.451**	1.45	**1.709**	1.7
23	84.574	**1.89**	1.86	**1.835**	1.83	**1.386**	1.38	**1.356**	1.35	**1.762**	1.71
23	81.61	**1.7**	1.66	**1.599**	1.57	**1.353**	1.35	**1.27**	1.27	**1.503**	1.5
24	99.267	**2.167**	2.1	**2.129**	2	**1.284**	1.28	**1.27**	1.27	**1.591**	1.6
24	95.407	**2.066**	1.99	**1.829**	1.81	**1.193**	1.19	**1.214**	1.21	**1.482**	1.48
24	91.925	**1.813**	1.79	**1.686**	1.82	**1.281**	1.28	**1.184**	1.18	**1.418**	1.4
25	98.717	**1.681**	1.62	**1.604**	1.61	**1.265**	1.26	**1.228**	1.22	**1.533**	1.53

25	96.388	**1.738**	1.73	**1.851**	1.82	**1.281**	1.28	**1.336**	1.33	**1.51**	1.51
25	98.77	**1.777**	1.75	**1.757**	1.75	**1.353**	1.35	**1.365**	1.36	**1.616**	1.6
26	98.681	**2.136**	2.4	**2.032**	2.4	**1.276**	1.27	**1.23**	1.23	**1.604**	1.62
26	99.53	**2.26**	2.2	**2.166**	2.13	**1.544**	1.54	**1.478**	1.47	**1.914**	1.9
26	96.62	**1.862**	1.85	**1.762**	1.79	**1.211**	1.21	**1.175**	1.17	**1.503**	1.5
27	96.185	**2.712**	2.66	**2.319**	2.34	**1.377**	1.37	**1.27**	1.27	**1.808**	1.8
27	89.58	**3.073**	2.95	**2.505**	2.5	**1.548**	1.55	**1.424**	1.42	**2.006**	1.99
27	98.732	**2.76**	2.71	**2.654**	2.62	**1.23**	1.23	**1.177**	1.17	**1.709**	1.709
28	93.613	**2.446**	2.44	**1.999**	2	**1.396**	1.39	**1.253**	1.25	**1747**	1.72
28	87.257	**2.236**	2.2	**2.1**	2.1	**1.26**	1.26	**1.168**	1.17	**1.672**	1.67
28	87.428	**2.019**	2	**1.767**	1.75	**1.122**	1.22	**1.061**	1	**1.461**	1.44
29	77.426	**2.144**	2.1	**1.937**	1.96	**1.544**	1.54	**1.415**	1.41	**1.808**	1.8
29	84.25	**2.05**	2	**1.704**	1.7	**1.362**	1.36	**1.195**	1.19	**1.529**	1.53
29	74.298	**1.949**	1.89	**1.624**	1.64	**1.396**	1.39	**1.124**	1.124	**1.503**	1.49
30	99.8984	**1.955**	1.95	**1.879**	1.9	**1.289**	1.29	**1.216**	1.21	**1.595**	1.59
30	99.916	**2.244**	2.2	**1.896**	2	**1.344**	1.34	**1.233**	1.23	**1.646**	1.65
30	99.876	**1.857**	1.84	**1.752**	1.76	**1.26**	1.26	**1.177**	1.8	**1.525**	1.51
31	74.137	**2.236**	2.2	**1.943**	2	**1.377**	1.37	**1.305**	1.3	**1.777**	1.77
31	73.59	**1.896**	1.9	**1.793**	1.8	**1.065**	1	**1.001**	1	**1.454**	1.45
31	82.7	**2.115**	2.1	**1.793**	1.8	**1.168**	1.17	**1.092**	1.09	**1.591**	1.54
32	98.889	**1.536**	1.53	**1.556**	1.59	**1.116**	1.12	**1.122**	1.12	**1.35**	1.35
32	99.111	**1.974**	1.9	**2.006**	2	**1.454**	1.45	**1.441**	1.44	**1.752**	1.74
32	98.006	**1.782**	1.7	**1.686**	1.69	**1.202**	1.2	**1.149**	1.15	**1.468**	1.47
33	83.515	**2.086**	2	**1.98**	1.99	**1.278**	1.28	**1.214**	1.21	**1.604**	1.6
33	97.678	**2.204**	2.2	**1.955**	2	**1.457**	1.46	**1.302**	1.3	**1.782**	1.74
33	96.319	**2.345**	2.3	**2.158**	2.15	**1.305**	1.3	**1.273**	1.27	**1.695**	1.7
34	99.524	**2.399**	2.38	**2.319**	2.31	**1.54**	1.54	**1.457**	1.46	**1.89**	1.86
34	92.413	**2.277**	2.2	**2.115**	2.1	**1.383**	1.38	**1.33**	1.33	**1.762**	1.7
34	88.912	**2.204**	2.2	**2.151**	2.13	**1.359**	1.36	**1.324**	1.32	**1.637**	1.63
35	87.637	**2.666**	2.6	**2.427**	2.37	**1.65**	1.65	**1.514**	1.51	**1.993**	2
35	78.019	**2.302**	2.25	**2.174**	2.11	**1.25**	1.25	**1.177**	1.18	**1.612**	1.61
35	86.373	**2.115**	2.1	**2.045**	2	**1.27**	1.27	**1.211**	1.21	**1.567**	1.56

The average accuracy of the algorithm's detection has been calculated:

Average Confidence = 89.67542%.

This indicates that the average detection of the system throughout the 105 test cases is almost 90% despite the challenging scenarios being chosen whereby in many cases, some potholes are barely noticeable to the naked eye. The experiment that took place was done within different times of the day, and within different weather conditions.

The average error distance of the system was calculated by subtracting the detected distance from the ground truth distance across all the different detections. The average of all these values was then calculated as per equation (4.9):

$$AverageErrorDistance = \sum\left(\frac{|(GT-ALG)_{TL}| + |(GT-ALG)_{BL}| + |(GT-ALG)_{TR}| + |(GT-ALG)_{BR}| + |(GT-ALG)_{C}|}{Total\ Test\ Cases}\right) = 0.008758667\text{meters} \quad (4.9)$$

Where: GT is the Ground Truth distance

ALG is the Algorithm-measured distance

TL is Top Left

BL is Bottom Left

TR is Top Right

BR is Bottom Right

C is Center

This indicates that the average error rate of the system is 0.00875 meters. The error distribution is illustrated in Figure 4.17.

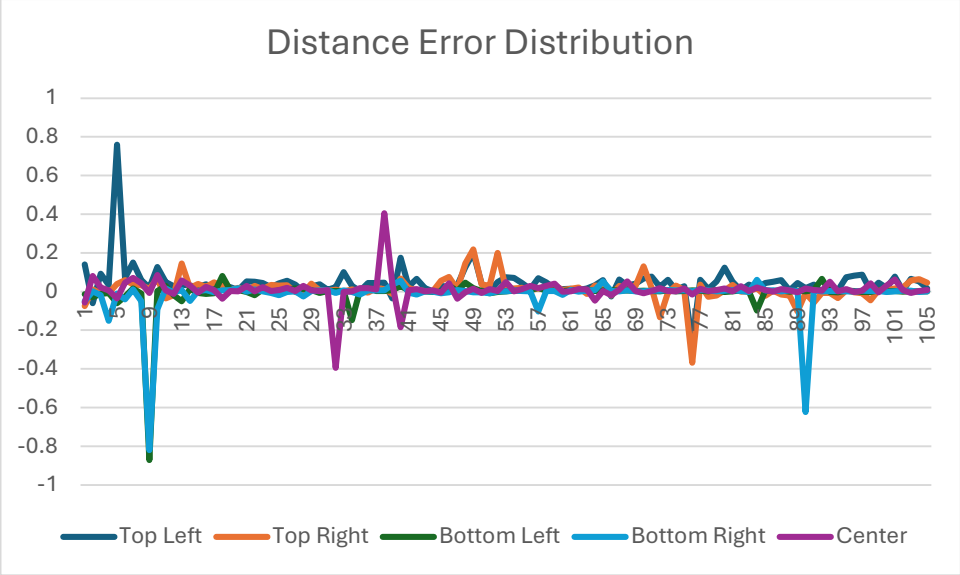


Figure 4.17: **Error distribution of the system's detection rate.**

It can be evident from Figure 4.17 that only two cases out of 105 test cases returned a result with a deviation less than 0.4 meters from the ground truth. This means that only two cases exhibited an undesired behaviour whereby the detected distance is higher than 0.4 meters from the ground truth. In the cases where the detected distance is higher than the ground truth distance, the result is still considered desirable as the avoidance prompts would be made earlier than the expected time, a behaviour that is still desirable as the pothole would still be avoided safely regardless of whether it was premature. Conversely, overestimations of the detection distance would trigger avoidance protocols beyond the optimal window, resulting in a missed opportunity to maintain a safe distance from the pothole at the designated threshold which is 1 meter.

The error rate has been calculated via the equation below:

$$Error\ Rate = \frac{2}{105} = 0.019047\ or\ 1.9047\% \qquad (4.10)$$

The minimum frame rate achieved within the measured detections is 816.92 Frames per Second.

By applying the minimum frame rate achieved within the Detection Rate (frames per meter,fpm) described earlier, the detection rate is calculated as follows:

$$Detection\ Rate\ \ 1 = \frac{816.92}{14.411} = 60.91\ fpm\ for\ a\ vehicle\ at\ 30\ mph \qquad (4.11)$$

$$Detection\ Rate\ \ 2 = \frac{816.92}{4.3333} = 245.07\ fpm\ for\ an\ EPW\ at\ 8\ mph \qquad (4.12)$$

$$Detection\ Rate\ \ 3 = \frac{816.92}{1.97} = 414.68\ fpm\ for\ an\ EPW\ at\ 4\ mph \qquad (4.13)$$

Notably, the system's minimum achievable frame rate of 816.92 fps surpasses the real-time detection threshold of 30 fps. This signifies that even under worst-case scenarios, the system can capture and process data at a rate sufficient for real-time obstacle detection. Furthermore, even when simulating wheelchair speeds comparable to car speeds in civilian areas (around 30 mph), the system exhibits a notable detection rate of 60.91 frames per meter. This high frame rate translates to a dense stream of information about the surrounding environment, exceeding the minimum required for reliable outcome prediction. Consequently, the system can achieve an error rate of less than 2%, demonstrating its ability to deliver accurate and trustworthy results for obstacle avoidance.

4.5 System's Limitations

Although the system has achieved a significantly groundbreaking and robust result, whether it is within the detection rate and accuracy, or the localisation and avoidance distance measurement, the system is still susceptible to certain limitations such as the extremely poor light conditions such as a path with absolutely no light as it relies on an RGB-D input feed whereby the depth feed would not return a reliable result without the RGB feed being able to perform well. In addition, ambient light, reflective surfaces, and shadows can impact depth accuracy, which is why the location of the camera is important to ensure optimal performance of the system. Furthermore, it is worth noting that the

minimal detection of the camera used is 0.5 meters which means that if the same camera were to be chosen, the system would not be ideal for any detections that are less than 0.5 meters, hence the safety distance being set to 1 meter.

Finally, the narrow field of view (84° horizontal, 53° vertical) of the camera might limit coverage in certain scenarios and could require the use of more cameras if the specified installation location cannot be used due to different reasons.

With all the mentioned limitations being discussed, the system is still considered reliable and most of the limitations can be mitigated by abiding by the set up instructions as they are, or by substituting the chosen camera with different models or technology that exceeds the Intel RealSense's performance.

Future work will focus on improving the system's accuracy by expanding the manually-collected dataset to include more examples of potholes from all around the world, not only the United Kingdom. This could potentially be done within a crowdsourcing project where individuals within the public can upload photos of negative road anomalies to expand the training dataset.

In addition, future work will focus on improving the versatility of the system by including less risky negative road anomalies such as cracks, patches, ravelling, and rutting. These anomalies, even though are not risky, but would cause certain discomfort to the user.

4.6 Conclusion

A groundbreaking technique for negative obstacle detection and localisation has been introduced within this chapter, leveraging the combined power of RGB photography and depth perception. This sensor fusion approach provides a comprehensive understanding of the environment, enabling the system to accurately identify and locate potential hazards, even those absent or poorly represented in traditional RGB images (e.g., potholes, hidden ledges, transparent obstacles).

The system's performance shines in real-time scenarios, achieving an exceptional minimum processing rate of 816.92 frames per second with an RGB-D Camera having an RGB frame rate of 30 frames per second, and a D (depth) frame rate of 90 frames per second. This rapid processing speed ensures near-instantaneous obstacle detection, crucial for timely avoidance manoeuvrers in dynamic environments. Furthermore, the system boasts a meager distance estimation error rate of 1.9047%, demonstrating its ability to pinpoint the location of hazards precisely with minimal margin for error. This accuracy is further underscored by the impressive mean average precision of 97.7%, highlighting the system's reliable detection of a wide range of negative obstacles.

Beyond its notable performance, the system is noteworthily scalable. Its reliance on readily available sensors minimises equipment requirements, making it easily mountable onto various moving platforms (e.g., electric-powered wheelchairs, mobility scooters, robots, and autonomous vehicles). This versatility enhances its applicability across diverse domains where obstacle avoidance is paramount. Additionally, the system's retainability further expands its potential. With minimal effort, it can be adapted to different environments and types of negative obstacles, ensuring adaptability and long-term utility.

In conclusion, this novel sensor fusion-based technique presents a significant leap forward in negative obstacle detection and localisation. Its exceptional performance, scalability, and adaptability position it as a promising solution for diverse applications demanding enhanced safety and precision in navigating dynamic environments.

Future work will be focused on including more negative road anomalies such as cracks and dips. It will also be focused on calculating the depth of the pothole in order to assess whether it can be traversed in the event where safe avoidance is unachievable and the only possible options are either to drive through it, or to engage in a full emergency stop.

Chapter 5

Incline Detection and Localisation via Point Cloud Imaging

5.1 Introduction

Assistive technologies are increasingly becoming a need for many individuals with disabilities regardless of what the disability is. One of the main heavily used assistive technologies is motor-related technologies such as electric-powered wheelchairs and mobility scooters. These technologies form an integral part of the user's everyday life; certain individuals might be full-time wheelchair or mobility scooter users depending on their disability. Others could be part-time users, the ones who have milder disabilities or chronic conditions. The number of registered wheelchair users in the United Kingdom according to the National Health Service, NHS [140] is 741,903 as of the second quarter of 2023 According to a study conducted by [141] in an average day, wheelchair users spend between 13.6 and 7.6 hours in a day on their wheelchairs.

For users who rely heavily or partially on assistive technologies, navigating within a physical environment often presents a complex interplay of barriers. Whilst key breakthroughs have been accomplished in the field of autonomous and semi-autonomous navigation of moving platforms, the accessibility infrastructure remains largely stagnant, with only minor to no improvements. In particular, the ubiquitous presence of wheelchair

ramps, inclined wheelchairs, accessible paths, dropped curbs, and other elevation changes remains a significant impediment and a never-ending challenge. Relying on visual cues, external assistance, or cumbersome manual manoeuvres requires a significant compromise on independence and confidence, casting an endless shadow over the promise of unimpeded mobility. This plants the seed for a new approach that centres around the presence of a novel system able to detect and assess traversable areas when faced with an incline or a decline, envisioning a future where individuals using wheelchairs can conquer such limitations with newfound autonomy and confidence.

This chapter presents a novel system that relies on depth data obtained from an RGB-D camera that uses multi-sensor fusion, combining stereo imaging with laser vision in order to generate an accurate point cloud describing the surrounding environment. The system processes the point cloud data in order to detect the inclined plane, whether it is upward-facing or downward-facing. Once the candidate plane is detected, the plane is then segmented in order to detect the dimensions of the plane and to split it from the normal ground. This is then followed by the estimation of the width of the plane, and its inclination angle. The system additionally factors in the offset ground inclination when pre-inclination is employed. Such pre-inclinations are used to mitigate the occurrence of sudden, steep slopes by generating a more gradual ascent/descent plane, improving manoeuvrability for wheelchair and mobility scooter users. If the width of the plane is larger than the assistive platform's width by a safe factor, and the inclination angle combined with the pre-inclination angle (if any) are within the legal limits of accessibility slopes designated by the local authorities and governing bodies, then the inclined plane is considered traversable. The resulting point cloud would be passed onto any autonomous or semi-autonomous navigation algorithm or simply returned as a visual guidance for the platform user in order to benefit from the assisted or manual navigation.

This chapter is organised as follows: statement of problem and system architecture are described in section 5.2. The results are discussed in section 5.3. The system's limitations are discussed in section 5.4, A real-life experiment to assess the system's performance is illustrated in section 5.5, and the chapter's conclusion is presented section 5.6

5.2 Methodology

5.2.1 Statement of Problem

The detection of inclined planes in a real-world environment is challenging in its nature due to many factors. The first factor lies in the fact that accessible ramps should be distinguished from bicycle ramps which have steeper slopes that cannot be traversed by wheelchairs, mobility scooters, or other assistive technologies. Although such ramps almost look the same as the accessible ramps, they should be clearly avoided as they could almost certainly result in falls potentially causing serious harm and injuries to the users. Accessible ramps should also be distinguished from naturally occurring inclines such as the ones found in parks and certain natural public areas. Furthermore, accessible ramps must possess adequate width to comfortably accommodate the passage of wheelchairs and mobility scooters, ensuring safe manoeuvring and preventing unintended contact with adjacent walls, or being at risk of impeded movement or potential obstruction by being trapped in between two narrow walls or edges. Moreover, the edges of the detected plane must be accurately segmented in order to ensure safe navigation throughout the ramp with no risk of falls in the case of ramps that have no borders, such as threshold ramps, folding ramps, and suitcase ramps. Such ramps typically have no railing, walls or any safety boundaries. For safety to be ensured, the system should be able to safely detect the edges providing the user or the navigation algorithm with enough information about the safety boundaries in order to avoid the risk of falls. Another important challenge is downward-facing inclines, where the user is located at the elevation terminus of the ramp with a downward-facing point of view. Consequently, limitations in the field of view of the imaging sensor and the user's visual perception may hinder the complete visibility of the inclined plane.

5.2.2 System Architecture

The proposed system comprises of the image acquisition module that relies on the Intel RealSense D435i RGB-D camera, and the Point Cloud Library [142]. The RGB-D camera scans the surrounding environment in order to generate the point cloud representation which is then fed into the Point Cloud Library API in order to generate the necessary detections and analysis. Exhibiting notable versatility, the RGB-D camera can be mounted on diverse platforms, from bicycles and mobility scooters to electric wheelchairs and even self-driving vehicles. To attain real-time processing and maintain low resource utilisation, and low complexity, the system focuses on the depth information provided by the RGB-D sensor, disregarding the RGB data stream. The input point cloud undergoes certain preparation steps. The initial data processing stage leverages a passthrough filter to refine the obtained point cloud. This filter selectively removes points with Z-coordinates outside the 0-2 meter range, corresponding to the expected ground plane and ceiling height based on sensor specifications. This targeted filtering eliminates approximately 30% of the data, primarily sensor noise and reflections, significantly reducing the computational burden and processing time for subsequent stages. Furthermore, it mitigates the influence of potential anomalies like stray points above the ceiling or noise below the ground, improving the accuracy and reliability of downstream detection algorithms. This initial filtering lays the foundation for further refinement through spatial and statistical filters, ensuring robust and accurate point cloud representation for subsequent tasks. After initial filtering, the Statistical Outlier Removal (SOR) filter is applied with refined parameters. While retaining the neighbourhood size of 50, the standard deviation multiplier is increased to 7. This adjustment effectively removes points whose distance from the mean distance within their neighbourhood exceeds seven standard deviations, identifying them as statistically significant outliers and subsequently filtering them out. Efficient processing with crucial information preservation is imperative for the system's successful performance; for this, the next step is focused on downsampling the filtered point cloud data. This is achieved by leveraging the voxel grid downsampling technique, where points within the predefined voxel size, or leaf size, of 0.07 meters are grouped together. This aims to balance the data reduction without sacrificing detail, mitigating potential processing delays due to large point cloud sizes whilst minimising information loss and maintaining high fidelity. This leads to a filtered and downsized ‘clean’ point cloud ready

to be processed by the algorithm. Following the downsampling stage, the analysis pipeline transitions to segmenting and analysing the identified inclined plane. This crucial step involves determining each point's surface orientation, or normal vector, within the point cloud. This is achieved with the help of the Tree search method with 50 nearest neighbours based on the average density of our point cloud detected.

The input cloud undergoes a segmentation process via the normal plane model estimation technique, This process aims to identify the dominant inclined plane where inliers reside close to each other. The Random Sample Consensus (RANSAC) is used as a robust iterative estimator aiming to estimate the plane inliers. For a candidate plane to qualify as an output plane, it must satisfy the strict criteria below:

Parallel Surface Normals: At each inlier point, the surface normal must be parallel to the surface normal of the output plane in order to ensure geometric consistency.

Angular Deviation Threshold: The maximum angular deviation between inlier point normals and the plane's normal must not exceed $\frac{\Pi}{2}$ or 90 degrees, preserving planar integrity.

Distance Threshold: Each inlier point must be within a 0.01-meter (1-centimetre) threshold from the segmented model, guaranteeing proximity and coherence.

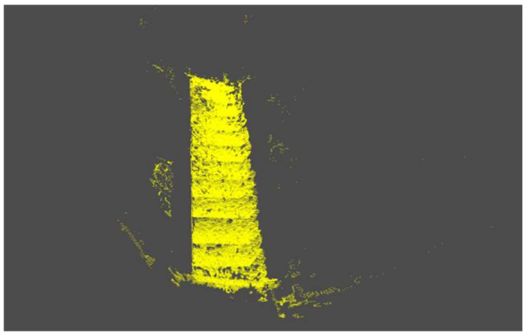


Figure 5.1: **Segmented Plane.** This figure represents the incline as it is seen after upon a successful segmentation.

Upon successful segmentation and identification of the region of interest (dominant plane), its estimated plane coefficients are extracted in Hessian form for further analysis and manipulation:

$$a.x + b.y + c.z - d = 0 \qquad (5.1)$$

The angle between the segmented plane and the ground is calculated with the help of the equation below:

$$C = Cos(\theta) = \left|\frac{\vec{n}\cdot\vec{P}}{|\vec{n}|\cdot|\vec{P}|}\right| \qquad (5.2)$$

$\vec{n}(0,0,1)$ *being the normal vector of the ground plane* $(\vec{x} = 0, \vec{y} = 0, \vec{z} = 1)$

$\vec{P}(a, b, c)$ being the normal vector of the segmented plane, where a, b, and c are the estimated coefficients returned by the algorithms in the hessian form as per equation (5.1).

By applying the coefficients estimated within equation (5.1) within equation (5.2), θ can be calculated within equation (5.3) as follows:

$$\theta = arcos(c)\frac{180}{\Pi} \qquad (5.3)$$

θ *being the inclination angle between the ground plane and the segmented inclined plane*

In addition to the inclination angle, θ' the pre-inclination angle, is recorded. The pre-inclination angle represents the angle of the small distance just before the slope change whether it is positive or negative. This distance serves as an intermediary surface that helps in avoiding sudden sharp changes in the slope when attempting to traverse an inclined plane. This angle is obtained via the Inertial Measurement Unit (IMU) of the RealSense camera bearing in mind an offset value equal to the tilting of the camera which is, in this case, 30 degrees, as per the tilting angle of the camera when installed on the electric-powered wheelchair used during the testing stage of the system. The pre-inclination angle is calculated with the help of equation (5.4):

$$\theta' = Pitch - 30 \qquad (5.4)$$

The inclination and pre-inclination angles are not the only important values for the safe traversal of the inclined plane. Another important aspect to measure is the width of the plane in order to ensure that it is wide enough for the moving platform to traverse it without the risk of colliding with the walls of the platform or, worse, being wedged between them. The measured distance should be greater than or equal to the chosen safe distance of 1 meter: knowing that the wheelchair's width is 60cm as per Figure 5.2, an additional 40 centimetres distance is added to take account for any unexpected errors.

Figure 5.2: **Width of the Electric-Powered Wheelchair.**

For the width of the segmented plane to be calculated with minimal-to-no errors, its concave hull is calculated. Due to its nature, being a polygon that encloses a set of points in a plane or an n-dimensional space, and unlike a convex hull, it can have concave angles and a smaller area, the concave hull can better capture the shape and boundary of a point set than a convex hull. For this, the alpha-shape value $\propto$ is set to 0.9 in order to estimate the most detailed, and the smoothest-possible representation of the edges of the segmented plane.

The estimated concave hull of the plane of interest is then used to calculated the coordinates of two key points of interest A(Xmin, Ymax, Zmin), and B(Xmax ,Ymax , Zmax) that represent the width of the plane at the closest point from the camera. The distance between both points A, and B is measured using the point cloud. This distance is then compared to the safe distance threshold of 1 meter, that is 40 cm, in addition to the width of the electric-powered wheelchair used to ensure safe traversal throughout the plane.

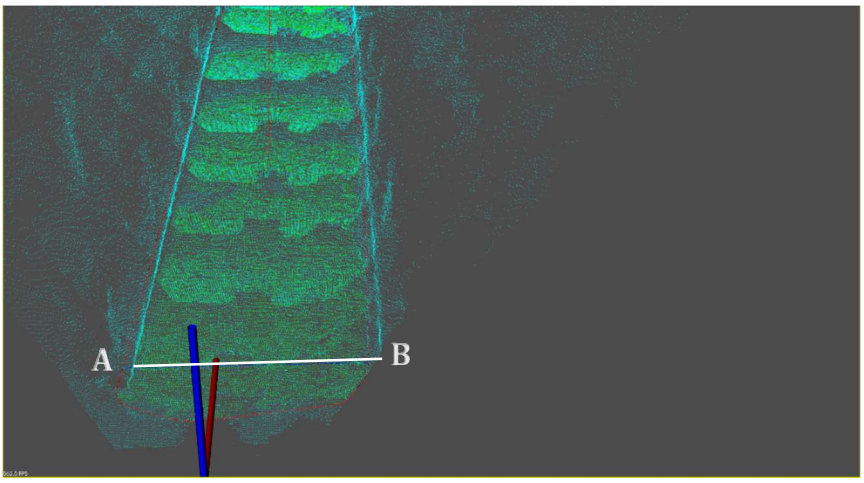


Figure 5.3: **A segmented plane(green), its concave hull (in red), and the estimated width of the plane (in blue).** The estimated width obtained by combining both points A, and B mentioned earlier is represented in a thin blue line and is measured in order to assess whether the plane is traversable or not.

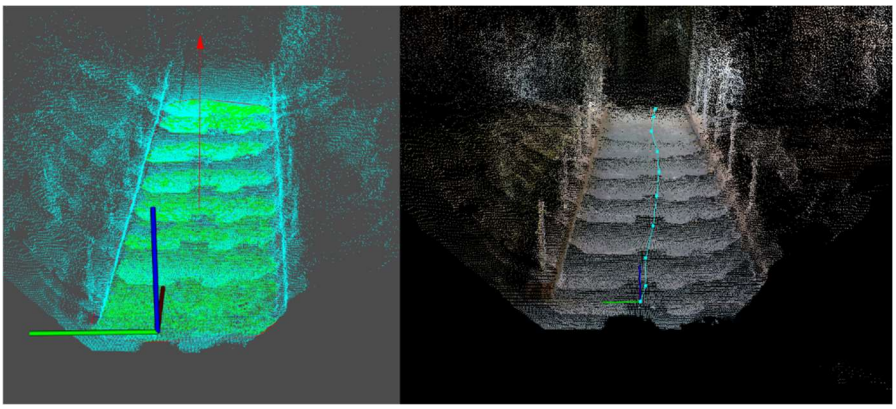


Figure 5.4: **The plane as observed by the RGB-D Camera (left) and its resulting segmented plane (right)**. This shows the plane as seen by the camera's RGB-D stream, and the resulting segmented plane represented with its normal vector (shown in red).

With the inclination angle, pre-inclination angle, and the width made available, the final step is to assess the traversability of the inclined plane. For an inclined plane to be traversable by a socially assistive technology, the plane's angle and inclination angle should be less than or equal to the maximum slope, or inclination angle allowed within the

local laws. In the case of electric-powered wheelchairs or manual wheelchairs, the maximum allowed slope for existing buildings with alteration difficulties is up to 1:12 (approximately 8.33 degrees) with additional safety measures like level landings at intervals and handrails on both sides, which means that the maximum inclination angle is 8.33 degrees for assisted wheelchairs used over short distances as per the British Standards 8300-1:2018. As mentioned earlier, the safety threshold for the width was set to 1 meter. Moreover, to account for minor errors, an additional safety offset of 0.33 degrees is deducted from the maximum inclination angle. This means that for the system to classify a segmented inclined plane as traversable, the inclination angle should be less than or equal to 8 degrees, and its width must be greater than or equal to 1 meter. Traversable planes are given the colour green, whilst unsafe ones are marked in red. Figure 5.5: **Wheelchair ramp as seen by the Intel RealSense D435i camera sensors** represents a wheelchair ramp as seen by the RGB-D camera's sensors. The figure illustrates how the wheelchair appears within the RGB Camera, and how the resultant depth feed appears along with the IMU readings. This view is passed onto the proposed system. The output is represented in Figure 5.6: The wheelchair **ramp from Figure 5.4 after segmentation and assessment. The** ramp is considered safe and marked in green as it has a width of 1.37 meters and an inclination of approximately 5.73 degrees. The system concluded that the ramp is around 1.37 meters wide and has an inclination angle of 5.73 degrees.

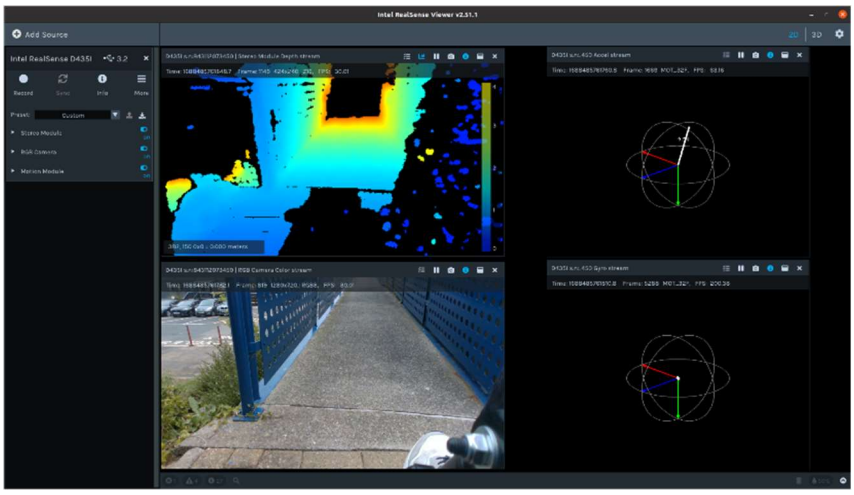


Figure 5.5: **Wheelchair ramp as seen by the Intel RealSense D435i camera sensors**

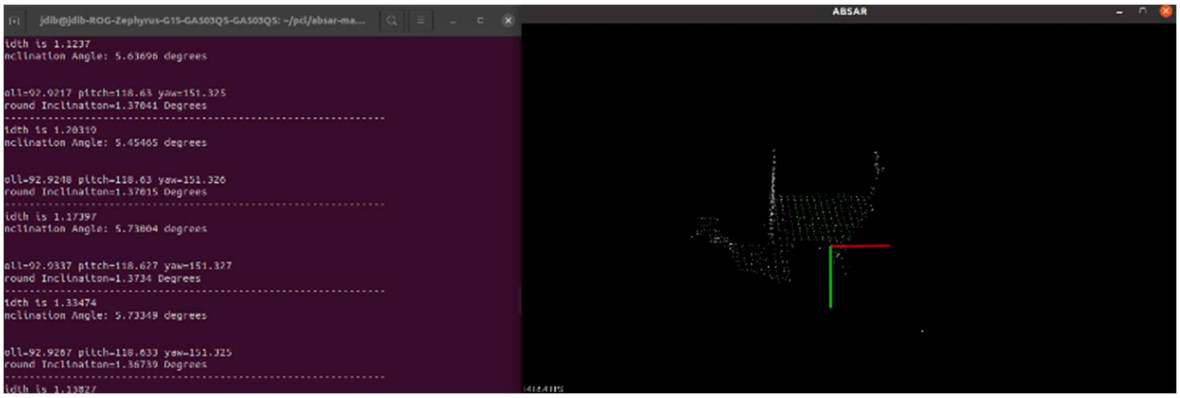


Figure 5.6: **The wheelchair ramp from Figure 5.4 after segmentation and assessment.** The ramp is considered safe and marked in green as it has a width of 1.37 meters and an inclination of approximately 5.73 degrees

Finally, to mitigate the impact of movement vibrations and any sudden changes in the camera's tilt, the system uses one of Intel RealSense's built-in calibration functions. The calibrations rely on the IMU to determine the rotation angles of the camera around all three 3D-Axis:

$$rotation = rotation_{\,x} \times rotation_{\,y} \times rotation_{\,z} \quad (5.5)$$

Where the rotations are calculated as follows:

$$rotation_{\,x} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & cos\,(\alpha) & -sin\,(\alpha) \\ 0 & sin\,(\alpha) & cos\,(\alpha) \end{vmatrix} \quad (5.6)$$

$$rotation_{\,y} = \begin{vmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{vmatrix} \quad (5.7)$$

$$rotation_{\,z} = \begin{vmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{vmatrix} \quad (5.8)$$

Where $\alpha, \beta, and\ \gamma$ are the counterclockwise rotation angles returned by the Intel RealSense's gyroscope otherwise known as pitch, yaw, and roll.

With Equation (5) calculated, the calculated rotation is used to generate the transformed cloud that constitutes the original point cloud redrawn based on an affine defined by an Eigen transformation matrix of all three rotations. This point cloud will constantly be generated whenever the pitch, yaw, or roll change making sure that the point cloud detected is not subject to any changes in these rotations. By doing as such, the resulting cloud will take into account any changes in the camera's rotation or tilt providing the system with the added capability to perform well even when the moving platform is exposed to movement vibrations or any other types of outside factors that could affect the camera's original tilt or performance. Consequently, the system culminates in a point cloud that remains invariant to changes in camera rotation, bolstering its robustness and reliability.

5.3 Experiment

To assess the system's performance and identify its limitations, a real-life experiment was set up using a Salsa M2 Electric-Powered Wheelchair (Figure 5.7: **Experiment setup.** The Salsa M2 Electric-Powered Wheelchair) as the chosen platform to be used within the trial process. The Intel RealSense D435i RGB-D camera was fitted on the left footrest upper hinge at an exact height of 53 cm from the floor with a tilt of 30 degrees (Figure 5.8).

Figure 5.7: **Experiment setup.** The Salsa M2 Electric-Powered Wheelchair

Figure 5.8: **Experiment setup**. Intel RealSense D435i fitted at 53 cm from the ground.

The wheelchair was driven around the University of Kent Canterbury Campus at different times and days. During the experiment, the estimated width of the detected inclined plane, and the estimated inclination angle of the plane were recorded along with the pre-inclination angle in the case of negative inclines. In addition, the decision of whether the inclined plane is traversable or not is also recorded. During this experiment, every wheelchair ramp was assessed from two different perspectives, the upwards angle where the wheelchair was located at the lower part of the ramp with the camera facing towards the ramp with an upwards perspective, and the downwards angle where the wheelchair was parked at the upper side of the ramp with the camera observing the

inclination from a downwards-perspective. Within this case, pre-inclination angle measurement also took place as pre-inclinations are usually introduced before downward-facing ramps in order to provide a pre-slope that makes traversing the ramp easier and safer. The EPW was also exposed to "dangerous" inclines that cannot be traversable, such as bicycle ramps with a significantly high slope. During the experiment, the ground truth for the width was measured with the help of a tape measure. The inclination and pre-inclination angles were measured with the help of a third-party mobile application named "Clinometer" by a company called SmartToolFactory [143]. The mobile application utilises the mobile phone's Inertial Measurement Unit (IMU) to measure ground inclination.

An additional experiment was conducted within the School of Engineering's Robotics Lab at the University of Kent. This experiment's aim was to assess the system's performance in an isolated environment, and to assess whether "Clinometer" which was used for measuring the ground truth is accurate of not.

During the second experiment, a ramp was built corrugated plastic sheet with a protractor and some firm supports that are used to create different inclination angle. The protractor was used in order to measure the accurate ground-truth inclination angle of the plane so that the system, and "Clinometer" can both be assessed accurately. Different angles were simulated starting 2 degrees up to 17 degrees (Figure 5.9).

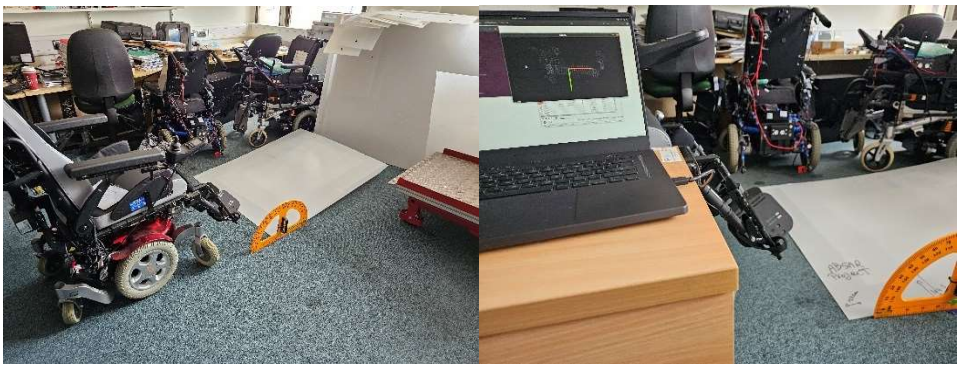


Figure 5.9: **Isolated environment experiment**. The wheelchair was parked in front of a manually built inclined platform with variable inclinations measured by a protractor in order to provide the ground truth information.

5.4 Results and Discussion

The resultant estimated width, inclination, and pre-inclination angles for the outdoor experiment are illustrated in **Error! Reference source not found.** along with their ground truths.

The average accuracy for the inclination angle, and the estimated traversable width estimation was calculated via the equations below:

$$Average\ Angle\ Estimation\ Accuracy = \sum \frac{Angle + Pre-inclination}{Real\ Angle} \times 100 \quad (5.9)$$

= 97.7142 %

$$Average\ Width\ Estimation\ Accuracy = \sum \frac{Width}{Real\ Width} \times 100 \quad (5.10)$$

$= 89.572\%$

c

Name of Incline	Angle	Pre-inclination Angle	Width	Real Angle	Real Width	Traversable? System	Traversable? Real
Biosc1	**5.95929**	**0**	**1.51047**	6	1.63	**YES**	YES
Biosc2	**3.73**	**2.2**	**1.5322**	4	1.63	**YES**	YES
Jennison1	**7.88**	**0**	**1.462**	7.2	1.5	**YES**	YES
Jennison2	**3.528**	**4.2**	**0.869**	4	1.5	**YES**	YES
Antro1	**6.25325**	**0**	**1.72661**	6	2.6	**YES**	YES
Antro2	**6.022**	**0**	**1.921**	6	2.6	**YES**	YES
lib1	**6.71555**	**0**	**1.84885**	7	2.6	**YES**	YES
lib2	**6.99947**	**0**	**1.96561**	7	2.6	**YES**	YES

libside1	**7.89807**	**0**	**1.51398**	8	1.9	**YES**	YES
libside2	**8.41011**	**0**	**1.52968**	8	1.9	**YES**	YES
libsidefirst	**7.33087**	**0**	**1.49056**	7	1.9	**YES**	YES
libsidefirst2	**5.15738**	**0**	**1.33629**	7	1.9	**YES**	YES
senate1	**7.31708**	**0**	**1.01528**	7	1.2	**YES**	YES
senate2	**7.19154**	**0**	**1.42423**	7	1.2	**YES**	YES
elliot1	**6.57773**	**0**	**0.585713**	6	1	**YES**	YES
elliot2	**4.48959**	**2**	**1.06041**	6	1	**YES**	YES
gymincline1	**6.68141**	**0**	**1.39841**	7	1.41	**YES**	YES
Gymincline1-2	**4.4744**	**2**	**1.1567**	7	1.41	**YES**	YES
gymincline2	**7.08282**	**0**	**1.38108**	7	1.4	**YES**	YES
Gymincline2-1	**4.47**	**2**	**0.94904**	7	1.4	**YES**	YES
gymside	**6.94269**	**0**	**2.19341**	7	2.4	**YES**	YES
gymside2	**3.28635**	**4**	**1.4589**	7	2.4	**YES**	YES
lawschool	**6.71481**	**0**	**1.44498**	7	1.2	**YES**	YES
Lawschool1	**4.62755**	**2**	**1.16812**	7	1.2	**YES**	YES
lawschool2	**5.0835**	**0**	**1.59744**	5	1.3	**YES**	YES
Lawschool2_1	**4.95316**	**0**	**0.586313**	5	1.3	**YES**	YES
footpath	**6.07014**	**0**	**1.50575**	6	2.3	**YES**	YES
footpath1	**3.41714**	**3**	**1.74725**	6	2.3	**YES**	YES
libback	**4.17975**	**3**	**1.06886**	7.22	2	**YES**	YES
libback1	**7.2022**	**0**	**0.7601**	7.22	2	**YES**	YES
rutherford	**6.89014**	**0**	**1.0961**	7	1.2	**YES**	YES
rutherford1	**5.09558**	**2**	**1.56582**	7	1.2	**YES**	YES
parkwood	**3.86911**	**3**	**1.63383**	7	1.7	**YES**	YES
parkwood1	**6.20163**	**1**	**2.00969**	7	2	**YES**	YES
parkwood2	**3.55286**	**6**	**1.32485**	6	1.6	**YES**	YES
parkwood3	**5.3335**	**4**	**2.44769**	6	1.6	**YES**	YES
bioscdown	**6.56107**	**0**	**1.262**	6	1.2	**YES**	YES
bioscdown1	**4.8091**	**1**	**1.52693**	6	1.2	**YES**	YES
librarygrimond	**5.20859**	**0**	**0.978582**	5	1.3	**YES**	YES
librarygrimond1	**2.06885**	**3**	**1.28856**	5	1.3	**YES**	YES
registry	**7.10328**	**2**	**1.73094**	5	1.7	**YES**	YES
registry1	**2.51276**	**3**	**1.03141**	5	1.7	**YES**	YES
registry2	**7.12527**	**2**	**1.19771**	5	1.5	**YES**	YES
registry3	**1.63226**	**3**	**1.64483**	5	1.5	**YES**	YES
rutherfordrampup	**5.73349**	**0**	**1.33474**	6	1.1	**YES**	YES
rutherfordrampdown	**5.83276**	**0**	**1.01953**	6	1.1	**YES**	YES
rutherfordramp2up	**5.57491**	**1**	**1.17654**	7	1.1	**YES**	YES
rutherfordramp2down	**5.68286**	**1**	**1.12606**	7	1.1	**YES**	YES
darwinbikeup	**9.01418**	**2**	**1.51953**	11	1.5	**NO**	NO
dawinbikedown	**9.06534**	**2**	**1.74813**	11	1.5	**NO**	NO
reject1	**10.4669**	**0**	**0.476506**	10	0.5	**NO**	NO
reject1down	**9.3599**	**1**	**0.48501**	10	0.5	**NO**	NO

The system achieved a commendable average accuracy of 97.7142% in estimating inclined plane width, and an average width estimation accuracy of 89.572%. The proportionally low average width estimation achieved can be attributed to various factors, including the camera's positioning relative to the plane, leading to perspective distortion, and partial occlusion of the surface by leaves or other objects, impeding accurate width determination. Importantly, all estimated width values remained below the ground truth, indicating a very low tendency towards false negatives (missing potentially traversable planes) but eliminating the serious and dangerous risk of false positives (misclassifying untraversable planes as usable). In order to mitigate the width detection discrepancy and to decrease the risk of false positives and negatives, an additional safe distance was applied to the safety threshold, as discussed earlier in this chapter. This significantly reduces the risk of false positives and negatives.

The average error distance was calculated as follows:

$$Average\ Error\ Distance = \sum\left(\frac{|GT-(P_{ALG}+\propto_{ALG})|}{Total\ Test\ Cases}\right) = 0.23094803 \qquad (5.11)$$

Where GT is the Ground Truth angle

ALG is the Algorithm-measured angle

$\propto$ is the Inclination Angle

P is the Pre-Inclination

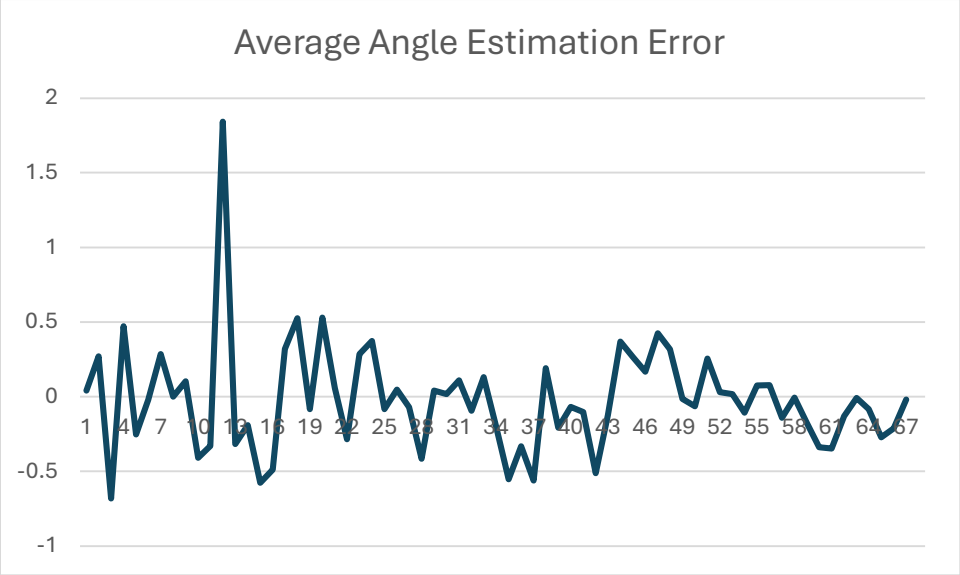


Figure 5.10: **Average angle estimation error**. This graph shows that only one instance out of 67 had a difference of more than 0.5 degrees from the ground truth.

As per Figure 5.10, only one instance had a difference of more than 0.5 degrees from the ground truth, hence, the error rate can be calculated as:

$$Error\ Rate = \frac{1}{67} = 0.01492\ or\ 1.492\% \qquad (5.12)$$

As for the isolated in-lab experiment, the results are illustrated in Table 5.1.

Table 5.1: **Second experiment results**. The table shows 16 different inclination angles simulated with the help of the platform built to imitate real-life scenarios. The table represents the angle and width estimated by the algorithm, the inclination angle measured by the "Clinometer" app, the ground truth measured with a protractor and a tape measure, and the binary of the traversability of the inclined plane.

Name of Incline	Angle	Pre-inclination Angle	Width	Real Angle	Real Width	Traversable? System	Traversable? Real
Lab1	**7.74426**	**8**	**1.06435**	8	1.04	**YES**	YES
Lab2	**6.97122**	**7**	**1.17131**	7	1.04	**YES**	YES
Lab3	**5.98218**	**6**	**1.39558**	6	1.04	**YES**	YES

Lab4	**5.10811**	**5**	**1.04523**	5	1.04	**YES**	YES
Lab5	**3.92468**	**4**	**0.811413**	4	1.04	**YES**	YES
Lab6	**2.92331**	**3**	**0.987194**	3	1.04	**YES**	YES
Lab7	**2.14143**	**2**	**0.988317**	2	1.04	**YES**	YES
Lab8	**9.00659**	**9**	**1.05934**	9	1.04	**NO**	NO
Lab9	**10.1735**	**10**	**1.02439**	10	1.04	**NO**	NO
Lab10	**11.3384**	**11**	**1.15517**	11	1.04	**NO**	NO
Lab11	**12.3482**	**12**	**1.03774**	12	1.04	**NO**	NO
Lab12	**13.1313**	**13**	**0.82224**	13	1.04	**NO**	NO
Lab13	**14.0093**	**14**	**1.13368**	14	1.04	**NO**	NO
Lab14	**15.0831**	**15**	**0.81986**	15	1.04	**NO**	NO
Lab15	**16.2719**	**16**	**1.05**	16	1.04	**NO**	NO
Lab16	**17.2117**	**17**	**0.830948**	17	1.04	**NO**	NO

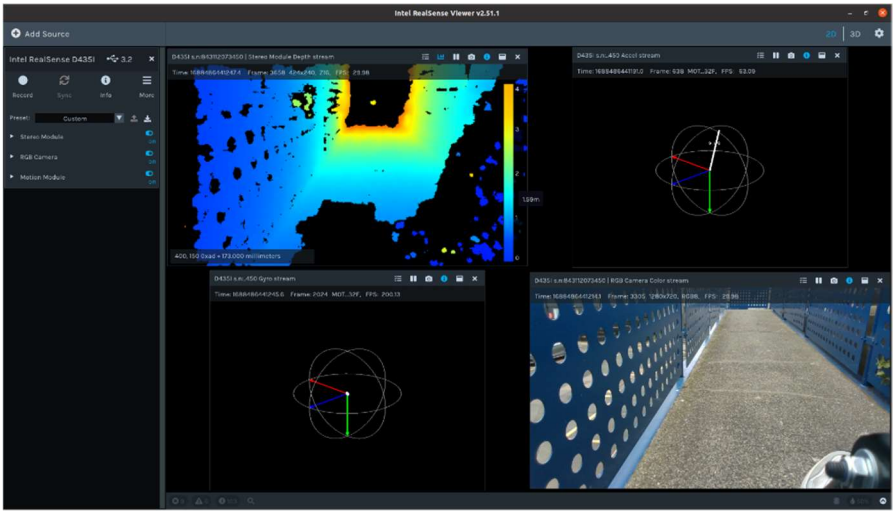


Figure 5.11: **"Rutherford" ramp as seen by the Intel RealSense D435i camera.** The inclined plane was perceived with an upward-facing perspective, labelled as “Rutherfordrampup” within ***Error! Reference source not found.***.

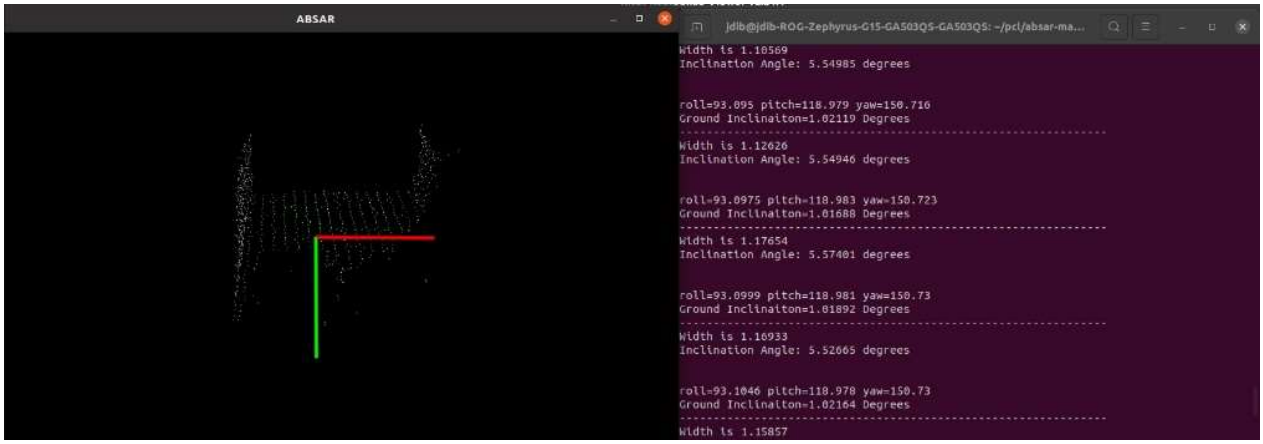


Figure 5.12: **"Rutherfordrampup" segmented (left) with its inclination and width estimated by the proposed algorithm (right)**. As the figure shows, the segmented plane was marked in green, indicating that it is traversable as its inclination is less than 7.2 degrees and its width is more than 1 meter.

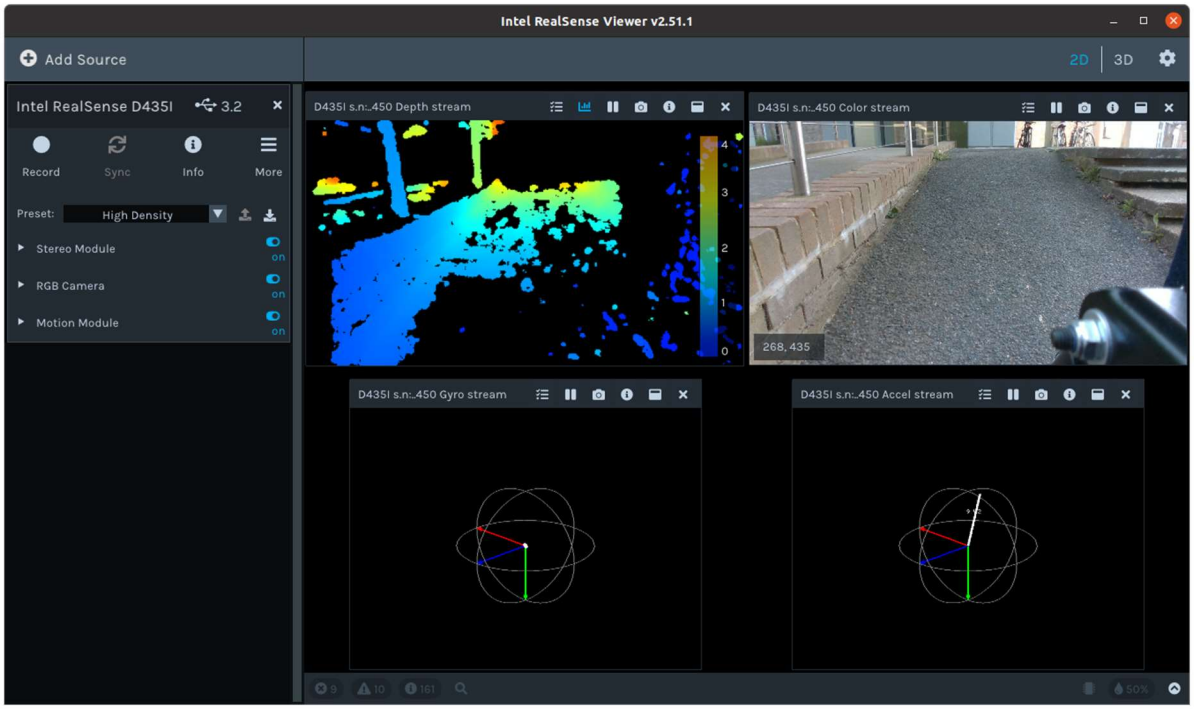


Figure 5.13: **Stiff bicycle ramp as seen by the Intel RealSense D435i Camera**. Labelled as “reject1” within ***Error! Reference source not found.***.

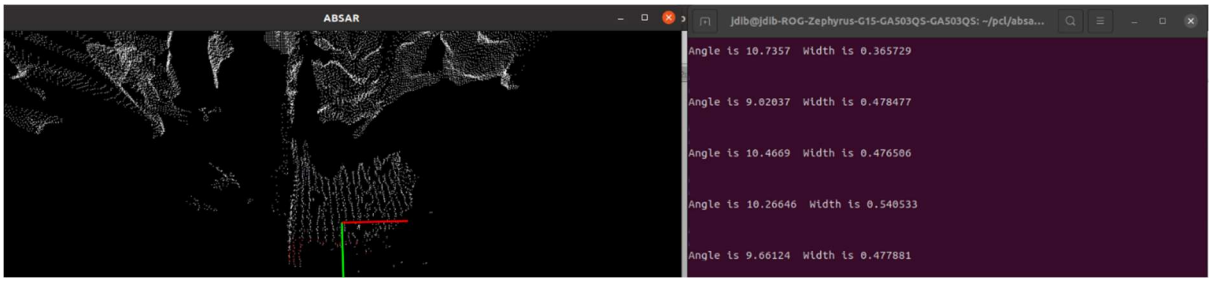


Figure 5.14: **"Reject 1" test results as returned by the proposed algorithm**. This figure demonstrates the segmented plane (left) and the algorithms' output (right). It is evident that the inclination angle is more than 7 degrees, and the width of the plane is less than 1 meter making it not traversable hence the colour red.

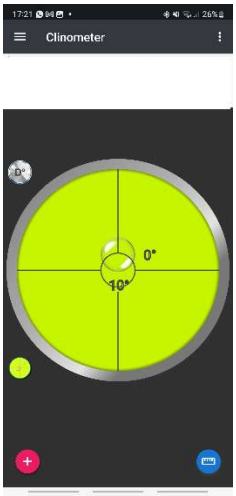


Figure 5.15: **"Reject 1" ground truth as measured by "Clinometer"**

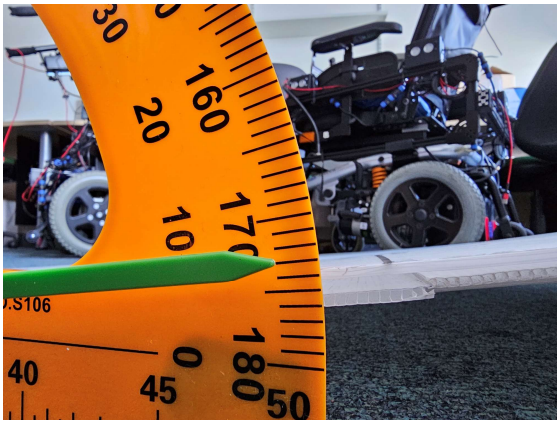


Figure 5.16: **"Lab 2" experiment instance within the indoors scenario as represented in Table 5.1**. This figure represents the ground truth as measured by the protractor.

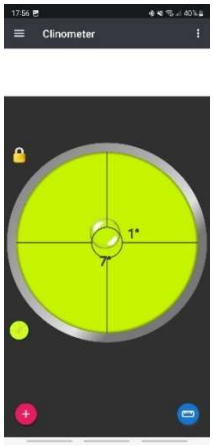


Figure 5.17: **Ground truth for "Lab 2" instance as measured by the "Clinometer" mobile application.** As illustrated, the inclination angle of the plane is 7 degrees, as measured by the protractor within Figure 5.15.

Figure 5.18: **"Lab 2" experiment instance output returned by the proposed algorithm**. The shows the segmented inclined plane (right), and the proposed algorithm's output (left) clearly shows that the estimated inclination angle is 6.97 degrees (ground truth is 7 degrees) and the estimated width is approximately 1.15 meters (ground truth is 1..04 meters), hence concluding that the inclined plane is traversable.

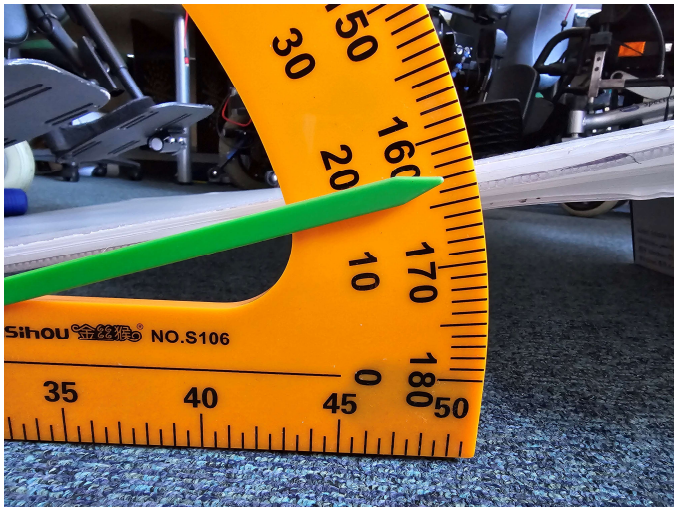


Figure 5.19: **Second experiment's "Lab 16" instance inclination angle as measured by the protractor**.

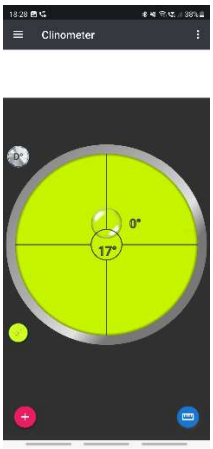


Figure 5.20: **"Lab 16" instance as measured by "Clinometer".**

Figure 5.21: **"Lab 16" Instance experiment result as returned by the proposed algorithm**. The figure represents the segmented plane (right) and the returned results (left). The estimated inclination as measured by the algorithm is more than 17 degrees, and the angle width is less than 1 meter, hence the ramp being considered as not traversable.

Figure 5.11 and Figure 5.12 illustrate the “Rutherfordrampup” experiment instance where the ramp near the Rutherford College at the University of Kent Canterbury campus

was measured. The estimated inclination angle is 5.73349 degrees, and the width is 1.33474. As the estimated inclination angle is less than 7 degrees, and the width is more than 1 meter, the ramp is considered traversable and is marked in green. This has been verified within the ground truth as the inclination angle is 6 degrees, and the width is 1.1 meters.

Figure 5.13 and Figure 5.14 on the other hand represent the "reject1", a bicycle ramp that is located next to the School of Biosciences within the Canterbury Campus of the University of Kent. The estimated inclination angle of the ramp is 10.4669 degrees, and the width is 0.476506 meters. The inclination angle is more than 8 degrees, and the width is less than 1 meter which makes this ramp not traversable. This has been verified within the ground truth measurements as the inclination angle is 10 degrees as per Figure 5.15 and the width is 0.5 meters.

Figure 5.16, Figure 5.17, and Figure 5.18 represent the "Lab2" indoors isolated experiment whereby an angle of 7 degrees was produced with the help of the protractor. The inclination of the angle was measured within the proposed system. The system's output clearly shows that the estimated inclination angle is 6.97122 degrees and the width is 1.17131. As the inclination is less than 8 degrees, and the width is more than 1 meter, the ramp is considered traversable and is marked in green. As for the ground truth measured by "Clinometer", and by the protractor, the real inclination angle is 7 degrees, and the real width is 1.04 degrees. The ramp is considered traversable as concluded by the proposed algorithm.

Figure 5.19, Figure 5.20, and Figure 5.21 represent the "Lab 16" indoors experiment whereby a ramp with an inclination angle of 17 degrees was produced. The algorithm's measured inclination is 17.2117 degrees, and the width is 0.950948. As the inclination angle is more than 8 degrees, the ramp is considered not traversable. This has been verified within the ground truth as the inclination angle as measured by the protractor and "Clinometer" is 17 degrees.

By leveraging equations (5.5) and (5.6), the average accuracy of inclination angle estimation reached 99.4785%, and the average accuracy of width estimation reached 94.003%. These results were obtained in a controlled environment with minimal external

interference, suggesting the system's potential for high reliability. Furthermore, the results demonstrate that the “Clinometer” mobile application can indeed be considered as an accurate ground-truth measurement tool for the experiments that were conducted.

A video is provided to showcase the system’s performance when traversing upwards facing icnlines. This video (video) is a demonstration of the BIOSC1 test case as represented in the Table 5.1. BIOSC2, a downwards-facing incline traversal demonstration is also illustrated within video (video). It can be observed within the video that the system is detecting the pre-inclination angle represented as Ground Inclination, and the actual inclination angle of the ramp itself. Pre-inclinations are used to minimise the impact of the downwards slope making the ramp easier and safer to access.

5.5 System’s Limitations

Despite the system’s significantly high performance and reliability in detecting, segmenting, and assessing upwards and downwards-inclined planes, the system is still prone to certain limitations, such as noise in low-light scenarios. Even though the system uses infrared imaging along with stereo imaging, it can still be affected by extremely low-light scenarios where certain false-positive points could appear within the detected point cloud. These points, if abundant, could potentially lead to the system falsely returning a width that is larger than the actual width of the plane which could lead to certain difficulties in traversing the system. Certain environmental factors could also impact the system’s performance. Such factors could include smoke, dust, and certain highly reflective surfaces. These could impact the quality of the resultant point cloud or, in certain occasions, cause obstruction to the camera’s viewpoint, leading to an empty or incomplete point cloud.

Finally, the Intel RealSense D435i requires a reasonably good processing device as it requires a relatively high power in comparison to other cameras. This is due to its

multimodal sensing technique and built-in IMU. For this, the system requires a medium-class processor to ensure the highest performance. This could potentially be considered a limitation depending on the moving platform chosen to host the system.

Although certain limitations could hinder the system's performance, these limitations remain minimal and uncommon. The system did prove its reliability in every test case within the conducted experiments making it a significantly good candidate for real-life implementations.

Future work and research would focus on investigating potentially expanding the sensory list to include an array of Time-of-Flight sensors fused with a LiDAR. This would significantly improve the system's reliability. One factor would still remain limiting is the system's computational requirements as it still needs to be able to be fitted onto a moving platform with minimal processing power, and with limited resources when it comes to electric power.

5.6 Conclusion

This chapter introduced a robust technique that enables autonomous, semi-autonomous, and manually-driven assistive technology platforms such as electric-powered wheelchairs, mobility scooters, or manually-driven wheelchairs to safely navigate around upwards-facing, and downwards-facing inclinations such as ramps and dropped curbs. The system leverages sensor fusion of RGB-D image acquisition systems and Inertial Measurement Units (IMUs) in order to capture a point cloud in the 3D environment. The system is not susceptible to surface vibrations or changes in its imaging sensor tilt due to its capability to recalibrate its resultant point cloud based on all possible 3D rotations.

A real-time accurate and highly-performing detection rate of 97.7142 % has been achieved with a significantly low error rate of 1.492%.

The proposed system is highly scalable as it solely relies on a core computing unit with medium-class computing power along with an RGB-D camera and an IMU. The system

boasts inherent adaptability, requiring no platform-specific modifications for integration onto diverse moving platforms. This scalability and adaptability render the system a versatile solution for various applications requiring robust navigation capabilities.

The system's performance has been subjected to a wide-spectrum experiment whereby it was tested in isolated and real-life environments, making sure that it can perform as expected within any possible scenario with high accuracy and a low error rate.

In conclusion, the proposed system presents a highly notable milestone that has been achieved, delivering enhanced safety to assistive technologies, making them more stable, and more autonomous, bringing the current technological advancements one step forward towards the ultimate goal of achieving a fully autonomous moving platform with no human intervention and minimal supervision.

Chapter 6

Curb and Downward-facing stairs Detection and Localisation

6.1 Introduction

Curbs and downward-facing stairs, seemingly unremarkable features of today's world's built environment, pose multifaceted challenges and are considered perilous obstacles for assistive technology users. Far from mundane, these ubiquitous features present complex navigation hurdles that necessitate a deeper understanding of their impact on mobility and safety. Examining these presumably simple structures through the lens of accessibility reveals intricate nuances. Curbs, intended to control traffic flow and delineate pedestrian areas, create sudden changes in elevation that can easily destabilise wheelchairs, increasing the risk of tipping or loss of control. Downward-facing stairs, designed for vertical movement, enabling individuals who do not rely on wheelchairs and other mobility-related technologies to easily and quickly change elevation, present an

insurmountable limitation for wheelchair users, effectively severing access to specific locations or even entire segments of the built environment.

Falls from these barriers represent a prevalent public health concern, inflicting physical injuries, hindering social participation, and incurring significant healthcare costs. Due to the lack of a specific National Health Service (NHS) database that records injuries caused by wheelchair falls in the United Kingdom, an alternative database, the National Electronic Injury Surveillance System 2003 in the United States, was accessed and it was found that it estimates that in 2003 [144], more than 102,300 individuals suffered injuries caused by wheelchair-related injuries. 65 – 80% of those injuries across all age groups are related to wheelchair falls [145]. Beyond the immediate physical danger, these obstacles have a profound psychological impact. The fear of falls from curbs and stairs can lead to anxiety, reduced confidence, and self-imposed limitations on mobility. This can translate to restricted participation in social activities, limited access to employment opportunities, and a diminished sense of independence. Additionally, these restrictive obstacles constitute an Achilles heel to the advancement of autonomous and semi-autonomous assistive technologies focusing on mobility. Their ubiquitous presence across various locations and scenarios makes them particularly problematic. They essentially represent the abrupt termination of a seemingly continuous pathway, often characterised by a sharp change in elevation. This inherent ambiguity makes accurate differentiation from traversable surfaces a hurdle for many detection techniques. Downward-facing stairs pose additional challenges due to their potential misidentification as ramps or inclined planes. Accessible ramps often share similar visual features, such as gradual inclines and smooth transitions, further complicating accurate detection. This difficulty in distinguishing between safe inclines and potentially dangerous drops presents a significant risk for wheelchair users.

Overcoming these challenges requires innovative approaches in curb and stair detection systems. Advanced image processing algorithms are crucial for discerning subtle visual cues and incorporating contextual information. Additionally, accurate spatial understanding is essential to accurately identify elevation changes and distinguish curbs and stairs from safe inclines.

This chapter addresses these limitations, by putting forward a robust and reliable traversable area detection system that empowers wheelchair users to navigate their environment with greater confidence and safety. This state-of-the-art environmental perception system utilises an RGB-D camera with built-in multi-sensor fusion techniques. By harnessing the combined strengths of stereo imaging and laser vision, the system generates accurate point clouds, offering detailed representations of the surrounding environment, enabling it to easily detect and assess curbs, downward-facing stairs, and other end-of-path edges by detecting the candidate planes, segmenting the traversable area making sure that the area is flat and unobstructed. This means that the system can provide an accurate estimation of the sharp change in slopes that could lead to severe harm. During the segmentation process, the system not only identifies the edge of the traversable area but also measures the safe traversable distance, making sure that the platform user is within the allowable safe distance as far away as possible from the risk of harm or injury. The system ensures that the user is constantly at a distance that is higher than the safety threshold set to make sure that all the users of such mobility assistive technologies are safe.

Moreover, the resultant point cloud would be used as an input to any autonomous or semi-autonomous assisted moving platform to enable it to move one step closer towards achieving the fully autonomous moving platform dream.

This chapter is organised as follows: statement of problem and system architecture are described in section 6.2. The results are discussed in section 6.3. The system's limitations are discussed in section 6.4, A real-life experiment to assess the system's performance is illustrated in section 6.5, and the chapter's conclusion is presented in section 6.6

6.2 Methodology

6.2.1 Statement of Problem

Due to the nature of the problem being addressed, traditional detection techniques cannot be considered. Downward-facing stairs, curbs, and the sharp ends of a path or road pose a significant limitation to most detection techniques widely used due to many reasons.

The first reason is visual ambiguity, as curbs and downward-facing stairs often blend in a seamless way within the surrounding environment, almost unnoticeable to the naked eye. This ambiguity, especially in poorly lit or uneven terrain, creates difficulties for existing detection systems based on image recognition. The second reason is the nature of existence of such obstacles. These sharp ends of different paths do not have a static warning sign of their existence, they could exist at the end of a grass-filled path, a tarmac-covered path, a cemented path, or even a tile-covered path in the case of downward-facing stairs. This renders techniques that rely on early detection unreliable. In addition to the previous, these obstacles usually have numerous and different characteristics. They exhibit considerable diversity in terms of size, shape, location, and materials. This poses a challenge for generic detection algorithms, requiring adaptability and context awareness. Not to forget the limitations of sensor technology. Traditionally-used sensors such as ultrasonic, and sonar-based systems have limited accuracy and range. This is clearly noticeable when such sensors struggle to differentiate between curbs, downward-facing stairs, and uneven surfaces. Furthermore, these sensors are highly susceptible to environmental factors like rain or dust, which can further hinder performance. Addressing the numerous challenges posed by curb and downward-facing stair detection is crucial for promoting safety, independence, and inclusivity for wheelchair and other mobility assistive technology users.

6.2.2 System Architecture

The proposed system encompasses the curb, and downward-stairs avoidance via the traversable area detection module. This module leverages the Point Cloud Library (PCL) as a foundational framework for analysing and processing Point Cloud data acquired via the Intel RealSense D435i camera. In order to overcome the visual ambiguity discussed earlier, the system exclusively relies on the on the depth information provided by the depth camera, disregarding the RGB colour data. This approach focuses solely on the estimation of the traversable area's shape, maximising computational efficiency, and minimising potential noise introduced by unnecessary colour information. The input depth data is processed through different stages in order to produce the expected results. The first stage consists of a passthrough filter that filters any noise out of the input cloud. For a point within the point cloud to be considered noise, its Z-coordinate must be outside of the [-2,2] range. Points that are not included within the threshold range are filtered out and removed from the input point cloud. After filtering the point cloud based on the Z-axis, the passthrough filter is reapplied to filter out points based on the X-axis with their coordinates not being within the [-5,5] range. This step ensures that the input cloud contains a limited amount of noise and that it lies within the focal length limitation of the camera. This also means that the segmented plane will have a width of 10 units, a distance that surpasses 1 meter, which is more than the width of most assistive technologies. Due to the fact that laser scans typically generate measurement errors that lead to numerous sparse outliers and irregular measurements and representations, the input point cloud must undergo an additional filtering stage. The input data is filtered with the use of a Statistical Outlier Removal (SOR) filter. This filter analyses each point's distance from its k-nearest neighbours (k=50 in this case) and compares it to the local standard deviation. Any point exceeding the mean distance by a predefined multiplier (8 in this case) is categorised as an outlier and subsequently removed. This approach effectively mitigates the impact of erroneous measurements, promoting a more refined and reliable point cloud representation for subsequent analysis. Following the filtering process, a parallel plane segmentation algorithm is applied to the input point cloud to isolate the ground plane. This

technique operates by identifying the pair of planes exhibiting the closest proximity while maintaining perpendicularity to the z-axis (representing the vertical direction). The algorithm seeks parallel planes, like the ground and a flat roof, that are close in proximity. The perpendicularity constraint to the z-axis ensures these planes are correctly aligned relative to the vertical direction. Due to this segmentation, distinct elevation changes (such as those representing curbs and initial downward stair steps) become readily detectable, enabling the system to differentiate the walkable ground plane from other elements in the environment.

Similar to other segmentation methodologies, 3D point cloud segmentation relies on identifying points of interest within a point cloud and grouping those that satisfy specific criteria into delineated objects or categories. This fundamental principle of segmentation applies across various domains, from image processing to robotics. In this instance, the target category comprises two parallel planes, both perpendicular to the Z-axis. Points residing within these planes are classified as inliers, adhering to the parameters of the segmentation algorithm.

To accomplish this plane detection, a robust model parameter estimator is essential. Estimators of this type fit a mathematical model to the point cloud data. Here, the Random Sample Consensus (RANSAC) algorithm provided by the point cloud library (PCL) is selected for its iterative nature and resilience to errors and outliers. RANSAC's iterative approach involves repeatedly sampling subsets of the point cloud, fitting a model, and evaluating the fit against the entire dataset. This allows it to effectively isolate the most representative inlier points in the presence of substantial noise or erroneous measurements. Additionally, RANSAC's computational efficiency is ideal for real-time applications. A distance threshold of 0.05 (5cm) ensures both the accuracy of the segmented planes and maximises point inclusion while preserving real-time processing capabilities. This threshold determines how tightly points must cluster around the model plane to be considered inliers. Choosing the optimal threshold involves a balance between precision and the need to accommodate some inherent point cloud variation. Upon the completion of the segmentation phase, one or two parallel planes would emerge as outputs

of this step. These planes are returned with their estimated plane coefficients that are represented in Hessian form:

$$a.x + b.y + c.z - d = 0 \qquad (6.1)$$

This step provides sufficient data to initiate the analysis and calculations phase that ensures a full understanding and assessment of the surrounding environment is fulfilled.

The subsequent stage involves the determination of specific points of interest (PoIs). The first is the centroid of the segmented planes, whose coordinates are derived from the previously introduced Hessian normal form. We denote this centroid as C (a, b, c). An additional PoI is selected as an arbitrary point in proximity to the centroid A ($\boldsymbol{x}_{\mathbf{max}}, \frac{\boldsymbol{y}_{min}+\boldsymbol{y}_{\mathbf{max}}}{\mathbf{2}}, \boldsymbol{z}_{min}$). The minimum and maximum points located within the Z-axis are recorded as Zmin and Zmax.

Finally, the distances between the camera and the previously determined points of interest are calculated. The distance between the camera and the centroid is denoted as DC, while the distance between the camera and the arbitrary point is denoted as DA. These distances are computed using the squared Euclidean distance equation:

$$Distance = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2} \qquad (6.2)$$

With $\boldsymbol{x_0} = \boldsymbol{y_0} = \boldsymbol{z_0} = \mathbf{0}$ as the coordinates of the origin point, and x,y and z being the coordinates of both points C and A, respectively. However, the distance between the centre of the camera and the centroid A is not enough as there are cases where the centroid could be very close to the edge of the traversable area, or in other cases, it could be mistaken with the centroid of the lower-side of the road that is situated under the curb. In order to avoid this, an additional measurement is introduced, this measurement value is named mD, or minSafeDistance is also calculated via the equation below:

$$\text{minSafeDistance} = \left|\frac{\text{CamHeight}}{\sin(180 - \text{CamTilt})}\right| \qquad (6.3)$$

with CamHeight being 0.5324 or 53.24 cm, and $CamTilt = \left(pitch \times \frac{180}{\Pi}\right)$. Pitch is provided by the IMU of the Intel RealSense D435i.

The detected plane is considered traversable if

(i) DC > minSafeDistance

(ii) minSafeDistance>1

(iii) Zmin < 1.97

(iv) Zmax > 1.

Zmin and Zmax are used to limit the traversable plane to the closest plane to the camera in the case where the camera observes a traversable plane and a curb.

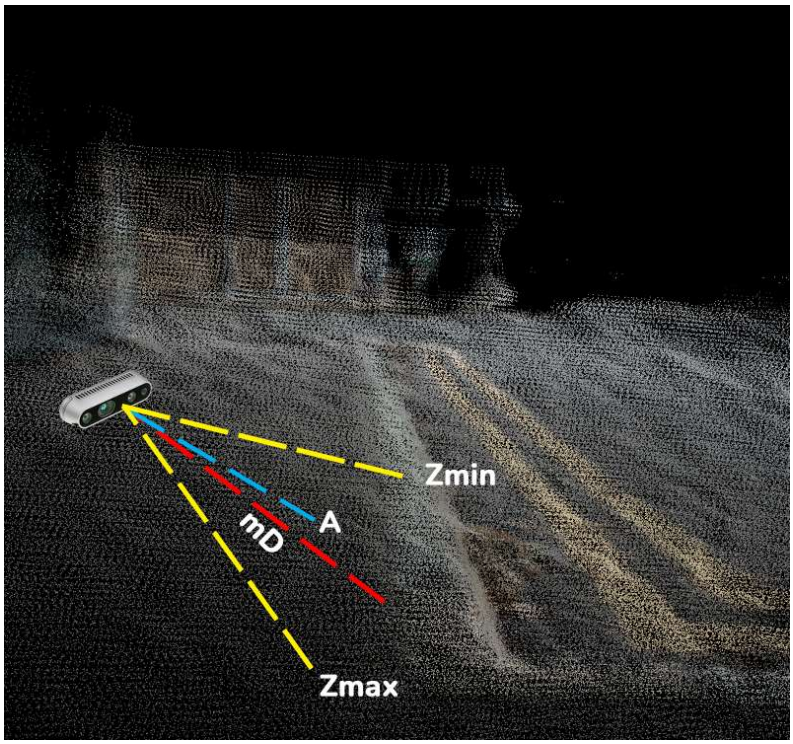


Figure 6.1: **Centroid A, Zmin, Zman, and mD (minimumSafeDistance).** Zmin and Zmax are used to accurately pinpoint the traversable area. mD is a safe distance that relies solely on the camera's

tilt and height to avoid cases where A is too close to the edge of the curb or A is mistakenly estimated as the centroid of the lower side of the curb.

If the detected plane satisfies the criteria for traversability, the system determines the traversable distance. This distance is calculated as the length between the camera and the plane's furthest point, while maintaining a predetermined safety margin of 1 meter. Conversely, if the plane fails to meet the specified traversability conditions, the system will generate a "STOP" message and associated command.

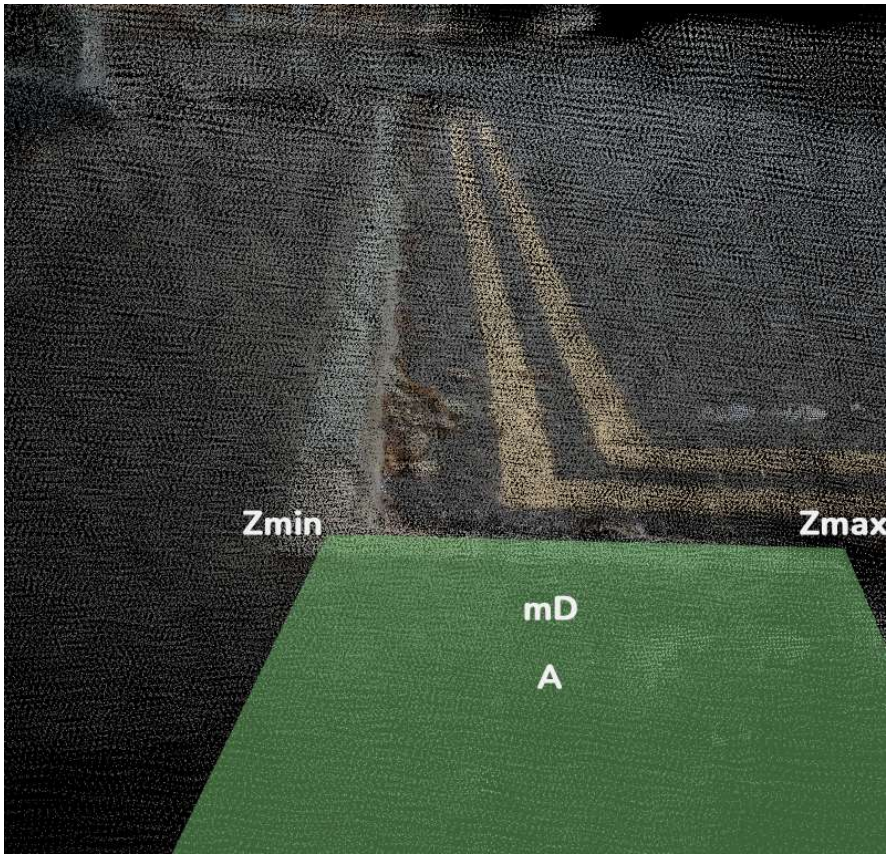


Figure 6.2: **Traversable area**. Illustration of what a traversable area would look like. The distance between the centre of the camera and Zmin, Zmax, A, and mD are measured and checked against the traversability conditions to ensure that the area in green is traversable.

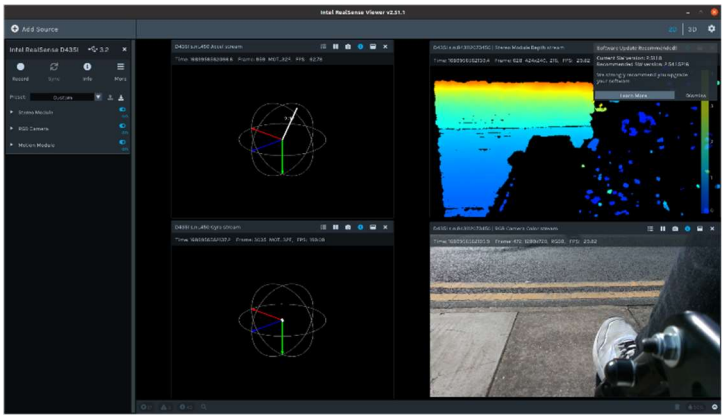


Figure 6.3: **Curb scenario as seen by the RealSense D435i camera.**

Figure 6.4: **Curb scenario after being assessed by the proposed method**.

6.3 Experiment

To rigorously evaluate the system's functionality and delineate its potential constraints, a field experiment was devised utilising a Salsa M2 Electric-Powered Wheelchair (Figure 6.5) as the primary platform for testing. An Intel RealSense D435i RGB-D camera was meticulously mounted on the left footrest's upper hinge, positioned at a precise height of 53 cm from the ground and angled at a 30-degree tilt.

Figure 6.5: **Experiment setup**. The Salsa M2 Electric-Powered Wheelchair with the D435i mounted onto the left footrest upper hinge.

Throughout the experiment, the wheelchair was navigated across various footpaths and curbs within the University of Kent Canterbury Campus, during different times of the day and in different weather conditions. The maximum traversable distance was measured and documented, with notations made upon encountering a curb or descending stairway. To establish ground truth, a tape measure was employed to precisely determine the distances from the camera to minSafeDistance, DC, and the overall Traversable Distance. The results are illustrated in Table 6.1. The system's point cloud processing rate achieved was 1350 FPS.

The average accuracy of the results was calculated via the equation:

$$Average\ Accuracy = \sum\left(\frac{\left|\left(\frac{GT}{ALG}\right)_D + \left(\frac{GT}{ALG}\right)_C + \left(\frac{GT}{ALG}\right)_{SD}\right|}{Total\ Test\ Cases}\right) \qquad (6.4)$$

$$= 0.999701723$$

The system's average error distance was also calculated. This involved subtracting the detected distance from the ground truth distance for each measurement. The resulting differences were then averaged, yielding an average error of 0.0034150 meters, as calculated using the following equation:

$$Average\ Error\ Distance = \sum\left(\frac{|(GT - ALG)_D + (GT - ALG)_C + (GT - ALG)_{SD}|}{Total\ Test\ Cases}\right) \quad (6.5)$$

Where: GT is the Ground Truth distance

ALG is the Algorithm-measured distance

D is the traversable distance

C is the Distance Center

SD is minSafeDistance

Table 6.1: **Experiment Results.**

Test Case ID	Distance	Distance REAL	DC	DC_Real	minSafeDistance	minSafeDistance Real
PEARS1	**2.22264**	2.22	**1.236543**	1.23	**0.516372615**	0.51
PEARS2	**1.27807**	1.23	**0.712293**	0.73	**0.61602974**	0.6
PEARS3	**1.3046**	1.13	**0.873954**	0.87	**0.54488172**	0.53
PEARS4	**STOP**	0.92	**1.58059**	1.574	**0.5302**	0.52
PEARS5	**STOP**	0.75	**1.58421**	1.58	**0.76358922**	0.76
SIBSON1	**1.18095**	1.18	**0.896952**	0.89	**0.8054079**	0.8
SIBSON2	**1.15564**	1.15	**0.867901**	0.86	**0.55701848**	0.54
SIBSON3	**STOP**	0.8	**1.0824**	1.08	**0.5217168**	0.51
SIBSON4	**STOP**	0.755	**1.09778**	1.1	**0.52912996**	0.53
SIBSON5	**STOP**	0.665	**1.05706**	1.05	**0.50950292**	0.5
JENSTAIRS1	**1.63388**	1.63	**N/A**	N/A	**0.594018**	0.59
JENSTAIRS2	**1.15823**	1.15	**0.893078**	0.89	**0.676326**	0.67
JENSTAIRS3	**1.57875**	1.57	**N/A**	N/A	**0.98117**	0.98
JENSTAIRS4	**1.23509**	1.23	**N/A**	N/A	**0.959871**	0.96
JENSTAIRS5	**STOP**	0.755	**2.05189**	2.05	**0.680821**	0.68
SPORTS1	**1.16524**	1.16	**0.769902**	0.77	**0.722119**	0.72
SPORTS2	**1.63563**	1.63	**0.67394**	0.69	**0.64089**	0.67
SPORTS3	**1.16023**	1.16	**0.77168**	0.77	**0.584358**	0.59
SPORTS4	**STOP**	0.705	**1.13042**	1.13	**0.685053**	0.68

SPORTS5	**STOP**	0.635	**1.07448**	1.07	**0.642584**	0.64
SPORTSPARK1	**1.0447**	1.04	**0.698315**	0.7	**0.586733**	0.6
SPORTSPARK2	**1.1328**	1.13	**0.713909**	0.7	**0.546109**	0.58
SPORTSPARK3	**1.15269**	1.15	**0.736138**	0.73	**0.588344**	0.6
SPORTSPARK4	**STOP**	0.7	**1.1873**	1.18	**0.5462**	0.57
SPORTSPARK5	**STOP**	0.68	**0.998157**	1	**0.540937**	0.57
KENEDYST1	**1.13189**	1.13	**1.02761**	1.02	**0.782063**	0.78
KENEDYST2	**1.81466**	1.81	**1.04778**	1.04	**0.532813**	0.57
KENEDYST3	**1.38437**	1.39	**1.3266**	1.32	**0.652921**	0.65
KENEDYST4	**STOP**	0.94	**1.10903**	1.1	**0.717108**	0.72
KENEDYST5	**STOP**	0.88	**1.13729**	1.14	**0.734616**	0.75
SIBSONST1	**1.60949**	1.61	**0.868737**	0.87	**0.69413**	0.7
SIBSONST2	**1.65768**	1.66	**0.79776**	0.8	**0.559229**	0.57
SIBSONST3	**STOP**	0.98	**1.2376**	1.23	**0.53512**	0.53
SIBSONST4	**STOP**	0.92	**1.37691**	1.37	**0.65368**	0.65
SIBSONST5	**STOP**	0.99	**1.29322**	1.29	**0.646958**	0.65
SIBSONFIRSTST1	**1.87634**	1.87	**0.942054**	0.95	**0.837664**	0.83
SIBSONFIRSTST2	**1.23737**	1.24	**1.02813**	1.03	**0.85112**	0.85
SIBSONFIRSTST3	**1.32985**	1.33	**0.655989**	0.65	**0.597375**	0.6
SIBSONFIRSTST4	**STOP**	0.75	**0.974753**	1	**0.728602**	0.73
SIBSONFIRSTST5	**STOP**	0.69	**1.81702**	1.8	**0.560262**	0.57
GRIMONDST1	**1.42437**	1.43	**1.17402**	1.17	**0.730168**	0.73
GRIMONDST2	**1.70371**	1.7	**1.56025**	1.56	**0.558409**	0.56
GRIMONDST3	**STOP**	0.93	**1.30105**	1.32	**0.900026**	0.9
GRIMONDST4	**STOP**	0.7	**1.44742**	1.45	**0.935804**	0.94
GRIMONDST5	**STOP**	0.86	**1.39197**	1.41	**0.53621**	0.55
GRIMOND1	**1.62911**	1.62	**1.35331**	1.35	**0.556353**	0.56
GRIMOND2	**1.53203**	1.54	**1.3249**	1.32	**0.544911**	0.55
GRIMOND3	**STOP**	0.96	**1.12884**	1.12	**0.72941**	0.73
GRIMOND4	**STOP**	0.99	**1.3263**	1.32	**0.812264**	0.81
GRIMOND5	**STOP**	0.86	**1.14381**	1.14	**0.810194**	0.81
JEN1	**2.11382**	2.1	**1.4123176**	1.41	**0.576650096**	0.57
JEN2	**1.50665**	1.5	**1.0064422**	1.02	**0.48510514**	0.48
JEN3	**1.30535**	1.3	**0.8719738**	0.87	**0.420291372**	0.42
JEN4	**STOP**	0.8	**0.5344**	0.53	**0.4575808**	0.45
JEN5	**1.11326**	1.1	**0.74365768**	0.72	**0.53659132**	0.53
GRIM1	**1.14841**	1.15	**0.76713788**	0.77	**0.55353362**	0.55
GRIM2	**1.6663**	1.65	**1.1130884**	1.1	**0.8031566**	0.8
GRIM3	**1.07769**	1.05	**0.71989692**	0.72	**0.51944658**	0.52
GRIM4	**STOP**	0.86	**0.57448**	0.57	**0.51542**	0.51
GRIM5	**STOP**	0.9	**0.6012**	0.61	**0.5338**	0.54

The system's average error distance was calculated throughout the entire experiment. The results are represented within Figure 6.6.

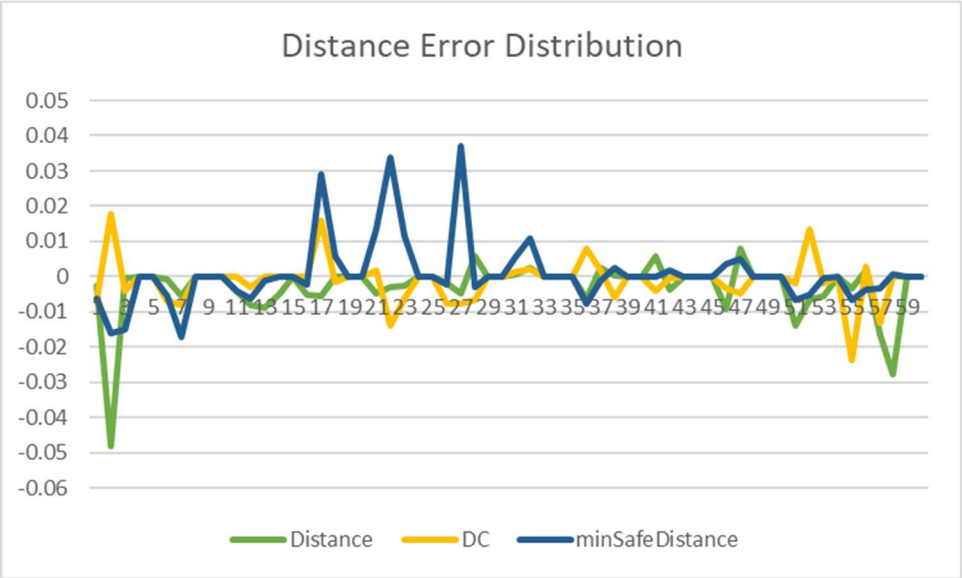


Figure 6.6: **Distance Error Distribution**. This figure describes the distribution of the distance estimation error throughout the conducted experiment.

Analysis of the table reveals that only two test cases exhibited a traversable distance discrepancy exceeding 0.01 m. Consequently, among the 60 test cases, only two instances demonstrated a measured distance deviating from the ground truth by more than 1 cm, with a maximum difference of 0.02769 m. Given the 1-meter safe distance threshold, this 3 cm measurement error is negligible. At worst, it represents less than 5% of the safe distance, ensuring the wheelchair will maintain a minimum 95 cm buffer from the curb.

Only a single test case out of 60 demonstrated a detection error exceeding 0.04 meters when compared to the established ground truth. This leads to a calculated detection error rate as follows:

$$Error\ Rate = \frac{1}{60} = 0.0167\ or\ 1.67\% \qquad (6.6)$$

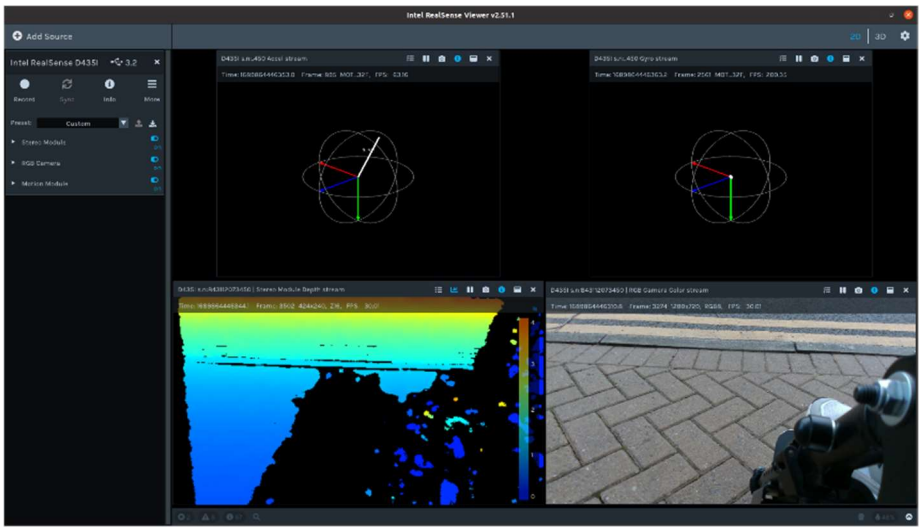


Figure 6.7: **Jen 2 test case as seen by the image input device sensor.**

Figure 6.8: **Jen 2 post-assessment as measured by the proposed algorithm**. The algorithm estimated the traversable distance to be around 1.50 meters and segmented the traversable plane based on the camera's field of view.

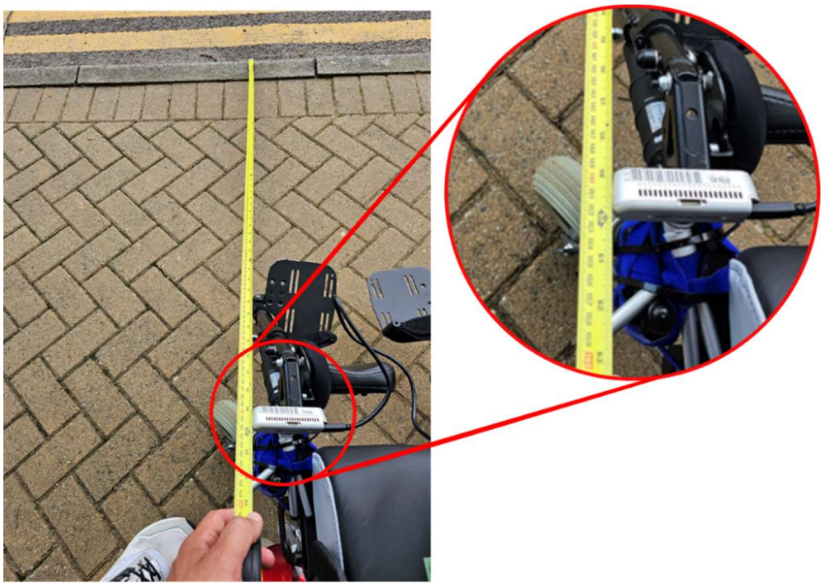


Figure 6.9: **Jen 2 ground truth**. As measured by the tape measure, it can be observed that the traversable distance is 1.5 meters.

Figure 6.7 illustrates one of the test cases recorded within Table 6.1, named Jen2. In this test case, the depicted scenario is one of the curbs that are situated near the Jennison Building within the University of Kent Canterbury Campus. It represents a normal curb in one of its simple forms of existence. The traversable distance as per the proposed method is estimated as 1.5 meters as illustrated in Figure 6.8, and as the traversable distance is more than 1 meter, the segmented plane is considered traversable and coloured in green. This has been verified Figure 6.9, the ground truth as measured with the tape measure is 1.5 meters which confirms the algorithm's findings.

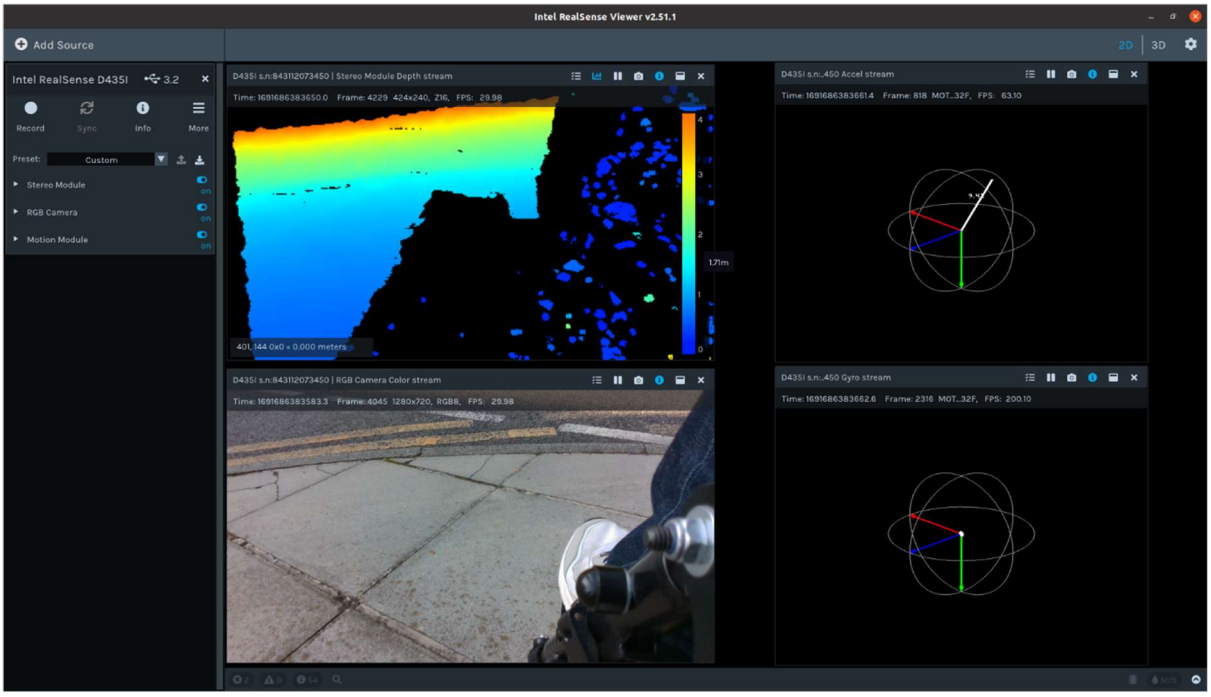


Figure 6.11: **GRIMOND 2 test case as seen by the imaging sensor**. As illustrated, the curb has a minor inclination and is not straight.

Figure 6.10: **GRIMOND2 test case as observed by the algorithm**. The camera is at 0.5324 meters from the ground, it is tilted at 44.66 degrees, the minSafeDistance is 0.544911, the distance centre is 1.3249 meters, and the traversable area is 1.53203 meters.

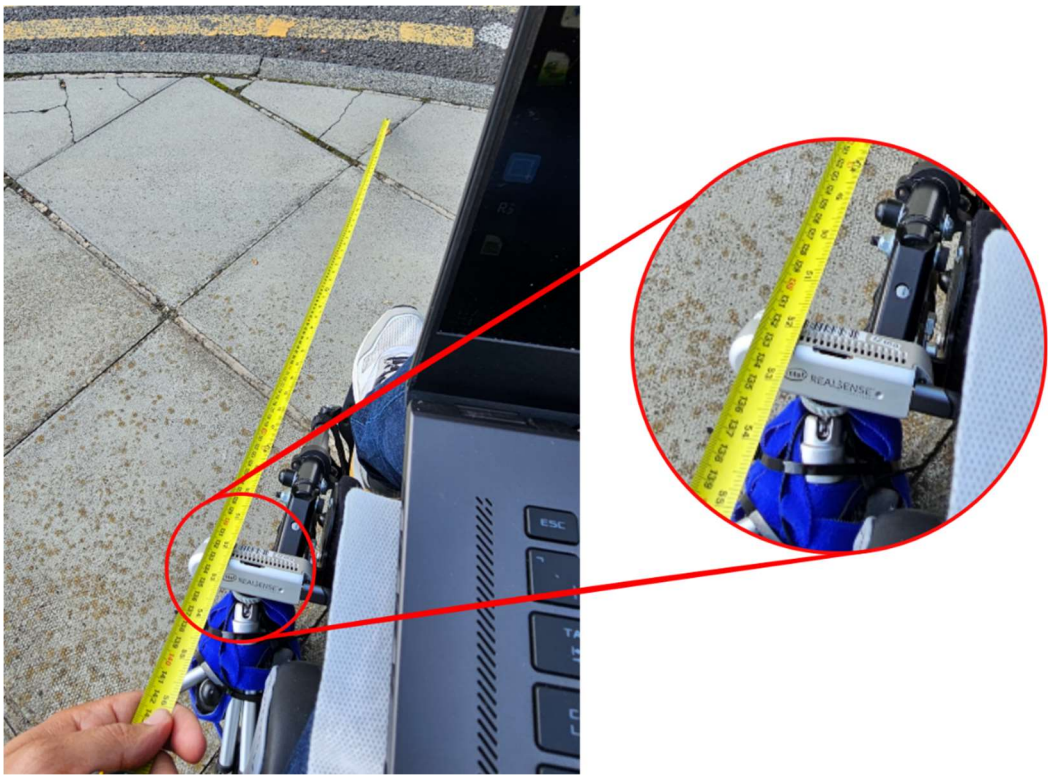


Figure 6.12: **GRIMOND2 - Distance Centre**. The ground truth for the distance from the camera sensor to the centroid of the segmented plane observed via the algorithm as measured by the tape measure. The measured distance is 1.32 meters.

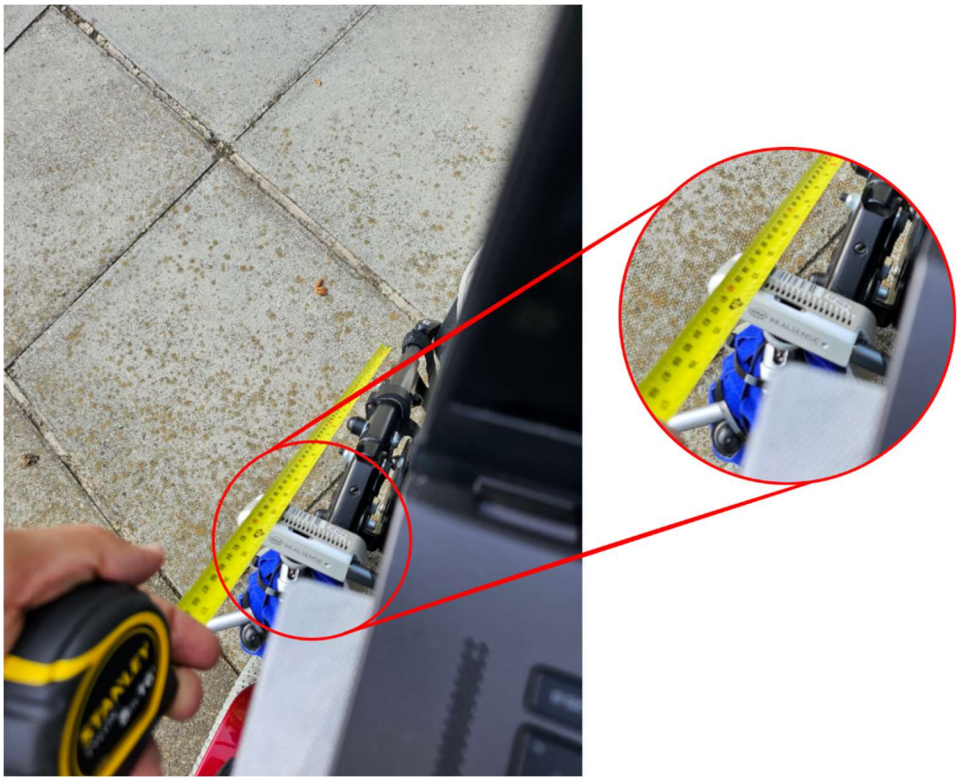


Figure 6.13: **GRIMOND2 Min Safe Distance**. The ground truth for minSafeDistance, the distance to be estimated via Equation (6.3) as measured by the tape measure. The distance is 0.555 meters.

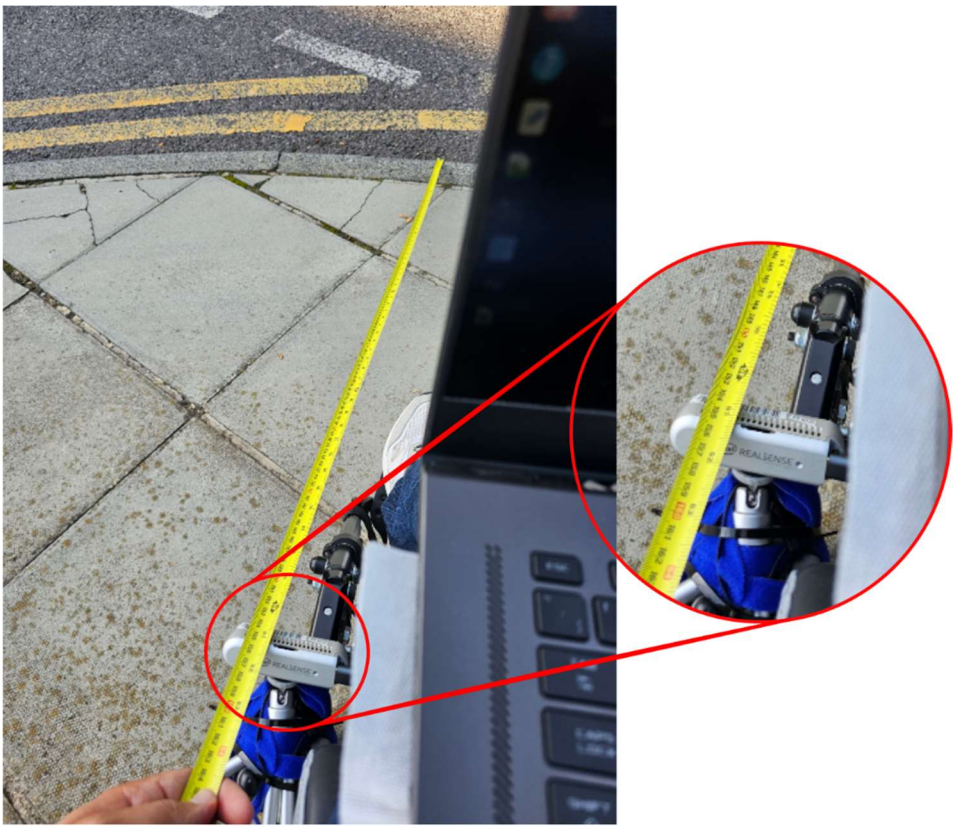


Figure 6.14: **GRIMOND2 Traversable Area**. The traversable area ground truth as measured by the tape measure. The distance is 1.54 meters.

Figure 6.10 illustrates another test case, GRIMOND2 recorded within Table 6.1.

GRIMOND2 is a curb located at the end of a semi-inclined ground surface. The electric-powered wheelchair and the input camera are located on the upper side of the curb, on the surface that is being assessed. Figure 6.16: KENEDYST1 **test case as observed by the algorithm**. The camera is at 0.5324 meters from the ground, it is tilted at 31.0599 degrees, the minSafeDistance is 0.78096, the distance centre is 1.02528 meters, and the traversable area is 1.13189 meters illustrates the algorithm's returned results after assessing the ground surface. According to the system, The camera is 0.5324 meters above the ground, its tilt is 44.66 degrees, which means that according to equation (6.3), the minSafeDistance is 0.544911 meters. The distance between the camera and the centroid of the segmented plane is 1.3249 meters whilst the traversable distance is 1.53203 meters. Upon assessing the ground truth with the use of a tape measure, Figure 6.12 confirms that the ground truth for the distance between the camera and the centroid is 1.32 meters. In addition, according to Figure 6.13, the measured ground truth for the minSafeDistance is 0.555 meters. As for

Figure 6.14, the measured traversable distance ground truth is 1.54 meters. The segmented area is more than 1 meter long, hence, the area is considered traversable and is marked in green. Upon comparing the ground truth to the estimated value, it can be evident that the system is highly accurate and reliable as the estimated traversable area is 1.53203 meters whilst the ground truth is 1.54 meters. This means that the error rate in this detection is 0.00797 meters, this enables the error rate calculation that is 0.00797 divided by 1.54. The result is 0.5175% which can safely be considered negligible.

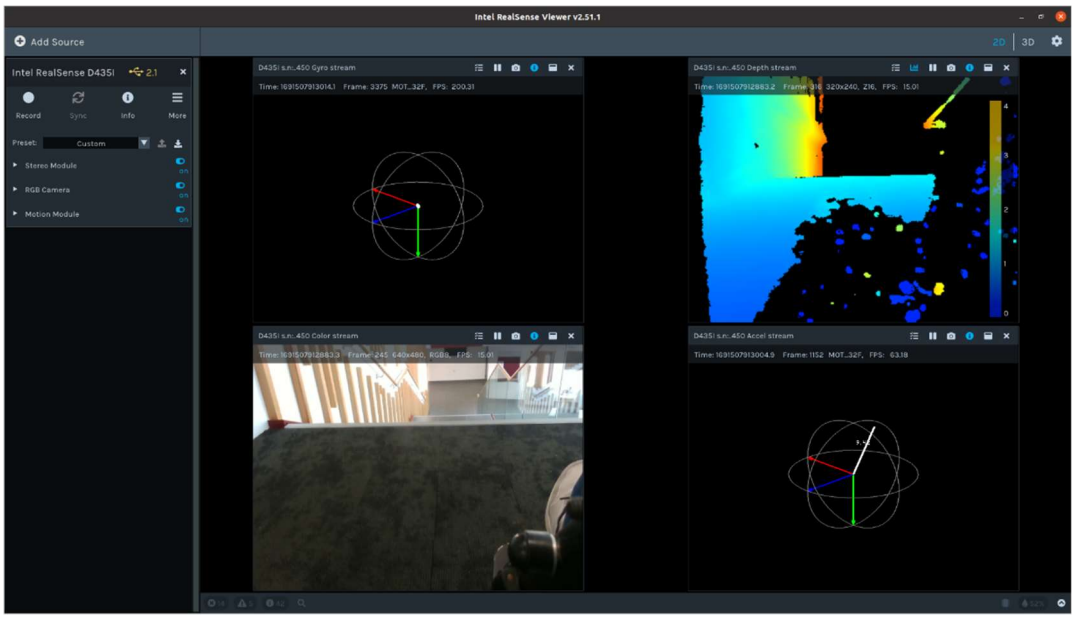


Figure 6.15: **KENEDYST1 test case as seen by the imaging sensor**. As illustrated, a downward-facing set of stairs located indoor within the Kenedy Building at the University of Kent Canterbury Campus.

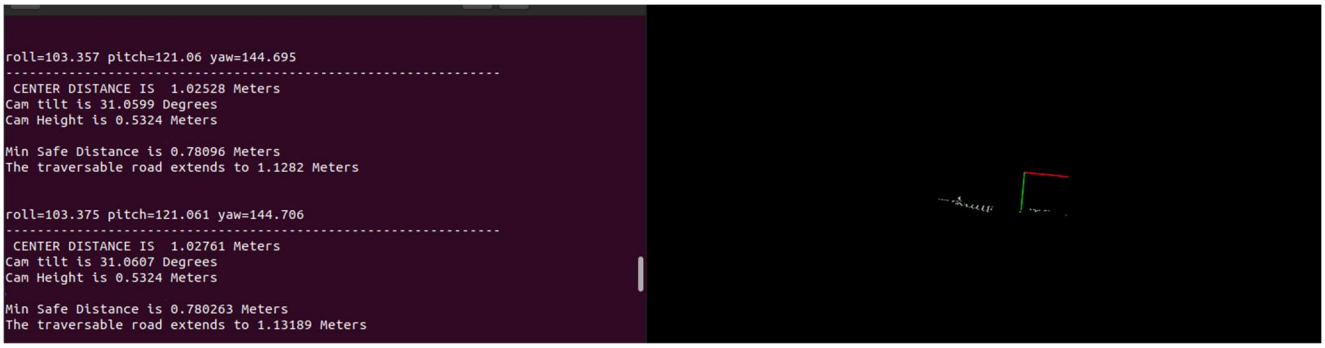


Figure 6.16: **KENEDYST1 test case as observed by the algorithm**. The camera is at 0.5324 meters from the ground, it is tilted at 31.0599 degrees, the minSafeDistance is 0.78096, the distance centre is 1.02528 meters, and the traversable area is 1.13189 meters

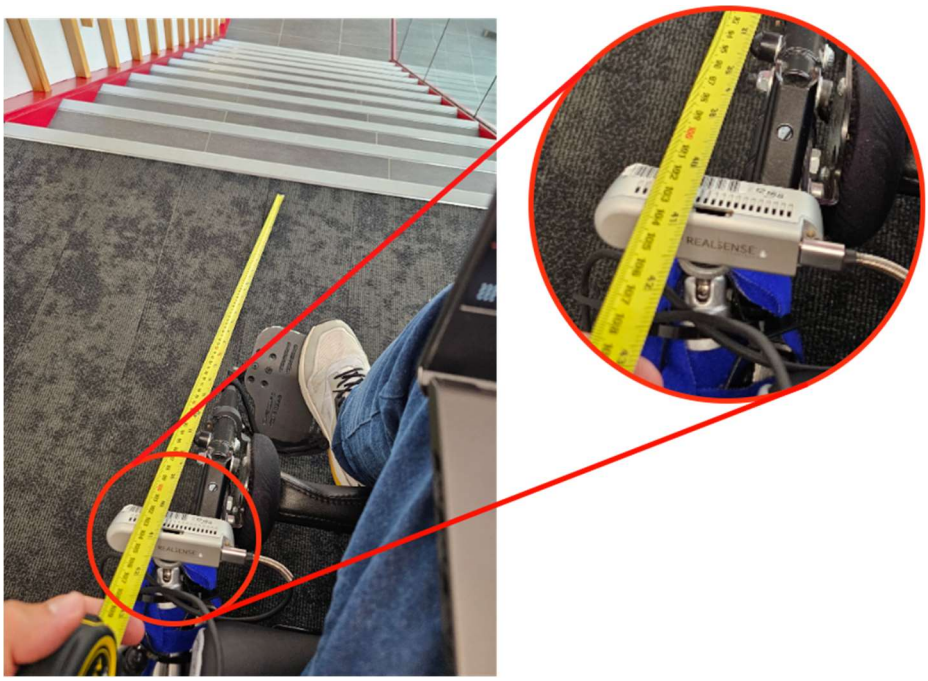


Figure 6.17: **KENEDYST1 - Distance Centre**. The ground truth for the distance from the camera sensor to the centroid of the segmented plane observed via the algorithm as measured by the tape measure. The measured distance is 1.02 meters.

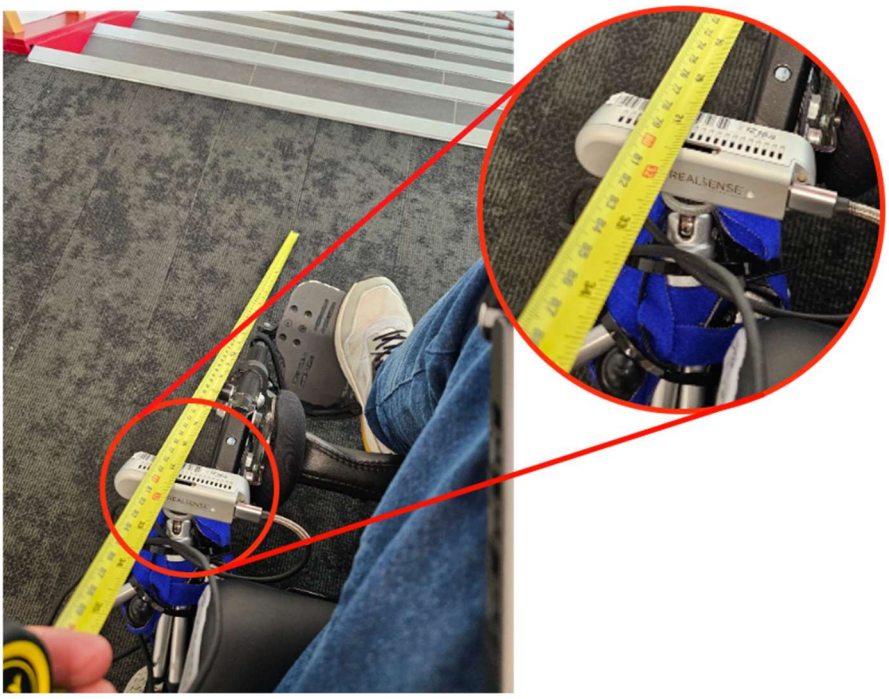


Figure 6.18: **KENEDYST1 Min Safe Distance**. The ground truth for minSafeDistance, the distance to be estimated via equation (6.3) as measured by the tape measure. The distance is 0.78 meters.

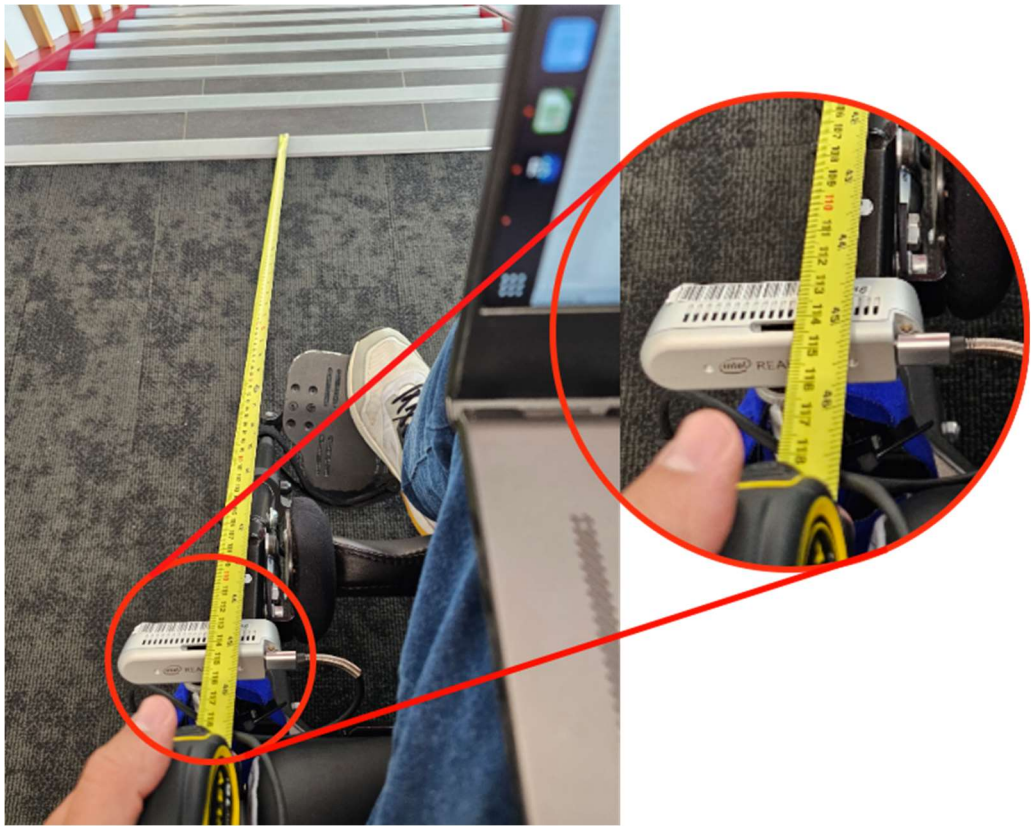


Figure 6.19: **KENEDYST1 Traversable Area**. The traversable area ground truth as measured by the tape measure. The distance is 1.13 meters.

Figure 6.15: **KENEDYST1 test case as seen by the imaging sensor**. As illustrated, a downward-facing set of stairs located indoor within the Kenedy Building at the University of Kent Canterbury Campus. illustrates another test case, KENEDYST1 recorded within Table 6.1.

KENEDYST1 represents a set of indoor downward-facing stairs located within the Kenedy Building at the University of Kent in Canterbury. The electric-powered wheelchair and the camera are both located on the top of the stairs, as per the previous example, with a tilt of 31.0599 degrees. According to the system, the camera is 0.5324 meters from the ground, which means that according to equation (6.3), minSafeDistance is 0.78096 meters. The distance between the camera and the centroid is 1.02528 meters, and the traversable area is 1.13189 meters. Upon assessing the ground truth by using a tape measure, it is observed that, as per Figure 6.17, it can be observed that the real distance between the camera sensor and the 1.02 meters. The minSafeDistance, as per Figure 6.18 is 0.78 meters

as calculated by equation (6.2). The segmented area is more than 1.13 meters long, which means that the area is traversable and that it is marked in green. The estimated traversable area is 1.13189 meters whilst the ground truth is 1.13 meters. The difference in calculation is 0.00189 meters. This enables the calculation of the error rate, which is 0.00189 divided by 1.13 meters, which yields to 0. 167% as error rate, which can be considered negligible.

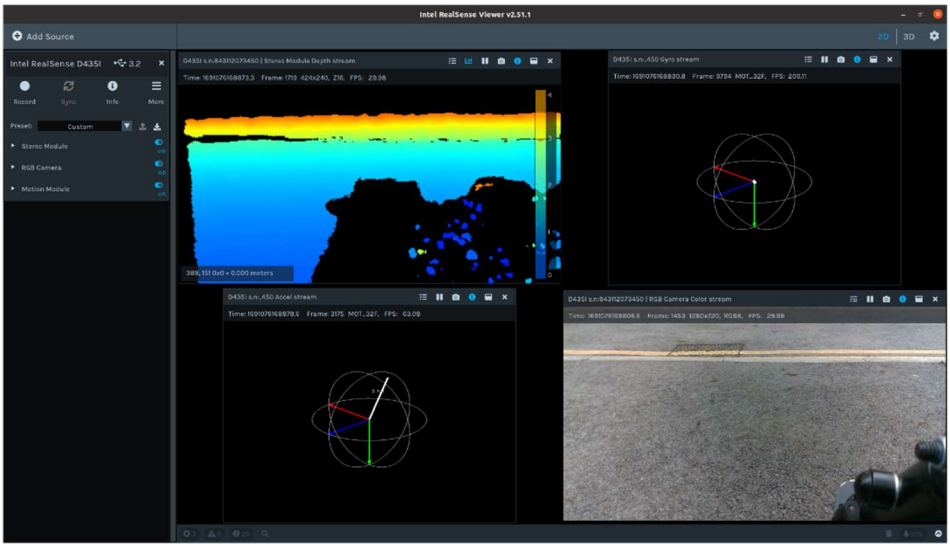


Figure 6.20: **PEARS2 test case as seen by the imaging sensor**. As illustrated, a curb located next to the Pears Building at the University of Kent Canterbury Campus was observed from a distance.

Figure 6.21: **PEARS2 test case as observed by the algorithm**. The camera is at 0.5324 meters from the ground, it is tilted at 31.0599 degrees, the minSafeDistance is 0.61602974, the distance centre is 0.712293 meters, and the traversable area is 1.27807

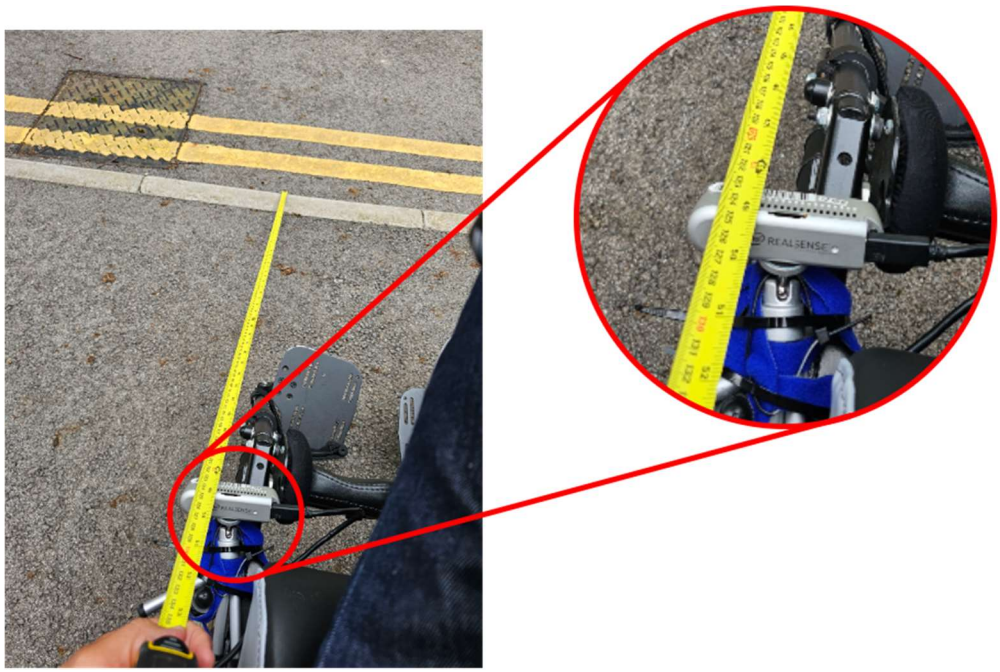


Figure 6.22: **PEARS2 Traversable Area**. The traversable area ground truth as measured by the tape measure. The distance is 1.23 meters.

PEARS2 represents a curb located outside of the Pears Building at the University of Kent in Canterbury. The electric-powered wheelchair and the camera are both located on the upper side of the curb, as per the demonstrated test case, with a tilt of 31.0599 degrees. According to the system, the camera is 0.5324 meters from the ground, which means that according to equation (6.3), minSafeDistance is 0.61602974. The distance between the camera and the centroid is 0.712293 meters, and the traversable distance is 1.27807 meters. Upon performing the ground truth assessment with the help of the tape measure, it is found that the real traversable distance is 1.23 meters, the distance to the centroid is 0.73 meters, and the minSafeDistance is 0.6 meters. It is obvious that the estimated distance is 0.04807 meters more than the real distance; this means that the error rate, which is 0.04807 divided by 1.23 meters, yields 3.9%. According to Table 6.1, PEARS2 is the test case with the highest difference from the ground truth, which means it is the worst-case scenario result. This scenario was detected with an error rate of 3.9%. As the safe distance threshold is set to 1 meter, and the minSafeDistance is 0.6 meters, the system will return the STOP command earlier than the detected traversable area becomes 0.04807

meters, this means that the user will be prompted to stop long before the anomaly in the detection is considered a risk to the safe navigation process.

6.4 System's Limitations

Due to the fact that the system uses stereo vision fused with infrared imaging, the system is still subject to a number of limitations. Despite employing both stereo and infrared vision, the system remains vulnerable to limitations inherent in optical sensing technologies. Dim lighting can compromise the integrity of the point cloud, hindering accurate curb detection. Infrared imaging provides additional fault tolerance by mitigating low-level scenarios where stereo vision fails. In addition to IR vision, the 1-meter safe distance threshold aims to mitigate any inaccurate readings by giving the user enough distance and time to intervene manually.

In addition, obscuring objects like debris or vegetation can disrupt the system's calculation of the traversable plane and could potentially provide false readings when it comes to the traversable distance or the "DC" or distance-centre value. This means that the traversable plane is augmented by merging litter or other objects with the ground level on the top side of the curb. This is where minSafeDistance plays a crucial role as it can detect a distance as the hypothenuse of the right triangle formed by the camera's tilt of 31.0599 degrees, and its height of 0.5324 meters would be, according to equation (6.3), 0.62151 meters. This means that the system will accurately determine when to stop more than half a meter away from the true end of the curb.

Future work will focus on improving the system's performance by investigating the use of Simultaneous Location and Mapping (SLAM) technology to better understand the surrounding when assessing ground planes and traversable areas. This would also enable the system to become more versatile and could potentially allow the system to navigate safely at a distance shorter than the 1-meter safe distance currently adopted.

6.5 Conclusion

This chapter presented a robust technique designed to enhance the safe navigation of curbs and downward-facing stairs for assistive technology platforms, including electric-powered wheelchairs, mobility scooters, and manually driven wheelchairs. This technique has potential applications in autonomous, semi-autonomous, and manual modes of operation. The system employs sensor fusion, combining RGB-D image acquisition and Inertial Measurement Units (IMUs) to generate robust 3D environmental point clouds. By incorporating IMU data, the system can dynamically recalibrate the point cloud to compensate for surface vibrations or changes in sensor tilt, ensuring accurate spatial representation.

The system exhibits a statistically significant detection accuracy rate of 99.97%. This is further underscored by a minimal error rate of 1.67%.

The proposed system demonstrates both scalability and adaptability. Its reliance on a standard computing unit, RGB-D camera, and IMU ensures minimal hardware requirements. This, combined with the system's platform-agnostic design, facilitates seamless integration onto various moving platforms without the need for extensive customisation. These characteristics make the system a versatile and cost-effective solution for diverse applications demanding reliable navigation.

In conclusion, the proposed system represents a significant advancement in assistive technology safety and autonomy. By enhancing stability and enabling greater autonomy, this work contributes to the ongoing development of fully autonomous moving platforms that prioritise user safety and minimise the need for human supervision. This system has the potential to expand the accessibility and functionality of assistive devices, improving mobility and independence for individuals with disabilities. Additionally, the principles and techniques demonstrated in this work could inform the development of autonomous systems across various domains, from self-driving vehicles to industrial automation. Further refinement and real-world testing will be crucial to ensure reliability and address any remaining challenges on the path to fully autonomous operation.

Chapter 7

Conclusion

7.1 Concluding Comments

This chapter provides a concise discussion of the research and its outcome. The incorporation of deep learning technologies along with machine vision has profoundly transformed the landscape of computer vision applications, fostering challenges and new lines of investigation, ushering in countless applications within different spectrums of life. One of the notable applications of deep learning and machine vision, whether they are fused or used separately, is the field of assistive technologies (AT). Harnessing the power of such technologies within AT can provide a significant improvement to the quality of life of its users, empowering more freedom of mobility and independence. One of the main limitations that AT users face is mobility and safe navigation of their mobile platform. For this reason, deep learning and machine vision technologies can potentially alleviate this challenge by providing guidance and assistance via spatial understanding of the environment in order to ensure safe navigation. This can be beneficial to manual, semi-autonomous, or autonomous navigation of assistive technologies.

This thesis introduces multiple algorithms based on deep learning machine vision techniques for detecting and localising negative road anomalies, as well as onboard computer vision point cloud segmentation and spatial understanding for upwards and downwards inclines, end of paths such as curbs, or downwards-facing stairs detection and localisation. The proposed systems do not require manual handling or intervention and are capable of achieving notable high performance and accuracy together with reliable real-time application.

The proposed systems are versatile and are not only limited to AT as they can be mounted onto any moving platform, whether it is a bicycle, motorbike, car, van, robotic platform, etc.

This thesis culminates in Section 7.2, where the key findings are presented and critically discussed. Additionally, Section 7.3 outlines promising avenues for future research and development.

7.2 Main Contributions and Research Findings

The answers to the research questions raised within the introduction are set out below:

- What impact does the variety of object representation within a negative road anomalies dataset have on the performance of the object detection network? Given that object detection systems are widely available using different technologies and setups, how can the optimal technique be selected?

Existing datasets are mainly focused on representing potholes from a car driver's or dashcam's perspective. They are mainly dedicated to on-the-road vehicles such as cars, vans, and other vehicles in an attempt to improve the field of driverless vehicles. These datasets often lack sufficient representation of water-filled, debris-filled, and diversely coloured. For this, a manually collected dataset was proposed in Chapter 3. The dataset is more aimed at potholes in their different forms and from different perspectives, usually the perspective of a pedestrian or an assistive technology user, whether a mobility scooter,

a wheelchair, or other ATs. The proposed dataset provides a wide coverage of potholes in different shapes, textures, and fillings, whether they are water-filled potholes, debris-filled, or rabble-filled, which bestows an integral role on the dataset for the successful completion of the desired application.

Chapter 2 provided a comprehensive review of the current literature, showcasing the different technologies used, their application, performance, and computational requirements. This comparison concluded that vision-based deep-learning techniques are the judicious choice for an accurate and reliable system designated for negative road anomalies or potholes detection and localisation.

Within Chapter 4, a comparison between three of the most prominent and widely used object detection convolutional neural networks was performed. The proposed dataset was applied to the chosen network, and the resultant detectors were compared. It was concluded that the most optimal network would be the one with the highest accuracy and least computational power requirements, as the network would need to be ported onto a low-performing processor with power limitations. Yet, the network is expected to perform in real-time with minimal to no latency. The resultant of the training process was represented and analysed, and the framerate performance was assessed via the application of the resultant networks onto a recorded video that serves as a real-time input to the network.

In conclusion, the manually collected dataset was essential for the designated task as it encompasses a wide variety of cases of pothole existence, taking into consideration the stochasticity of its nature and condition. As for the best-performing network chosen, the Tiny YOLO (You Only Look Once) convolutional neural network was chosen due to its high mean average precision, or mAP of 97.7%, and a significantly high detection rate of 532.99 frames per minute.

- How can deep learning-based object detection networks be utilised to accurately detect and classify potholes of varying sizes, shapes, and textures under diverse environmental conditions (e.g., lighting, weather, road surface) and in real-time?

Chapter 4 illustrates the process of choosing and training the most optimal convolutional neural network for pothole detection and localisation within a given video feed. After rigorous comparison and analysis, one network was chosen. The chosen network was then cascaded within a larger system where the input is not only an RGB feed but also a depth feed. RGB feed was applied to the chosen pre-trained network, and the resultant video frame was aligned with the depth feed in order to extract the measurement of the distance between the camera and certain points of interest within the real 3D world. The presented novel approach is a breakthrough in negative object detection and localisation. It enables the system to accurately provide precise distance readings that enable any negative obstacle to be treated as a positive obstacle when using the output as sensory data for any autonomous or semi-autonomous navigation algorithm. In addition, the proposed system provides accurate real-time feedback to the user, which enables AT users to safely navigate their way around potholes and negative obstacles whilst mitigating the risks associated with falls.

The system underwent rigorous testing designed to simulate a wide range of challenging real-world scenarios. These included adverse weather conditions, variations in time of day, and operation at high mobility speeds. Despite these challenges, the system demonstrated exceptional performance in the detection of potholes. It achieved a remarkably high confidence rate of 89.67542% in real-time operation. Furthermore, the system exhibited outstanding precision in distance measurement, with an average error of only 0.008758667 meters. These results offer compelling evidence that negative obstacle detection is feasible. Moreover, they validate the potential of treating negative obstacles as positive entities when utilising systems like the one presented.

- How can the geometric properties extracted from point cloud data be harnessed to detect and assess the accessibility of positive and negative inclined planes, including wheelchair ramps, in real-time scenarios?

Chapter 5 introduces an advanced computer vision system that leverages a unique fusion of depth camera data and Inertial Measurement Unit (IMU) readings to generate a comprehensive point cloud representation of the environment. This system is expressly

designed to detect, segment, and evaluate the characteristics of both upward and downward-inclined planes. Central to its operation is a sophisticated suite of filtering and segmentation algorithms, which precisely isolate the inclined planes of interest. The system then rigorously calculates the plane's inclination, taking into account the pre-inclination angle—a subtle change in the ground slope often preceding negative inclinations. Finally, a careful assessment of the segmented plane's width determines its traversability. The proposed system can mitigate sudden changes in the camera's tilting that could be caused by the exposure to continuous vibrations or other sudden shocks as it uses the IMU data to automatically calibrate and normalise the resultant point cloud in its real-life representation. This provides an accurate and reliable depiction of the real-world environment surrounding the platform hosting the system.

The proposed system achieved a noteworthy detection accuracy rate of 97.7142% with a significantly low error rate of 1.492%. The system does not require any substantial computing power and is capable of running on almost any processing unit that is capable of running a real-time point cloud representation of the surrounding environment.

In conclusion, the collected geometric properties obtained via an RGB-D camera can be used within the proposed algorithm to segment an inclined plane and obtain its plain coordinates in Hessian form. The coordinates are then used to calculate its inclination angle. This, along with additional distance-related and pre-inclination related measurements can be fused in order to detect and assess the traversability of inclined planes such as wheelchair ramps, curb drops, and other accessibility-related inclined planes.

- How can a point cloud segmentation algorithm be designed to efficiently and accurately segment curb and downward-facing stair edges in real-time scenarios within the real world?

Chapter 6 provides a comprehensive exploration of the design and implementation of a novel point cloud segmentation algorithm. This algorithm leverages a unique fusion of RGB-D camera data and Inertial Measurement Unit (IMU) readings, alongside innovative calculation techniques, to achieve robust segmentation and assessment of ground planes.

The system commences its operation with precise segmentation of the ground plane, followed by a rigorous traversability assessment. It detects the clear, traversable area leading up to the plane's termination point, where a curb or downward-facing stairs typically present a potential hazard.

To provide the user with vital situational awareness, the system meticulously calculates crucial measurements. These include the minimum safe distance and the total traversable distance from the camera's perspective to the plane's edge. By equipping the user with this precise information, the system facilitates the manoeuvres necessary for effective obstacle avoidance and fall prevention. Furthermore, this system holds the potential to serve as a valuable input for autonomous navigation algorithms, opening up new avenues for the safe and independent operation of assistive technologies.

The system demonstrated notable performance, achieving a noteworthy detection accuracy of 99.97% while maintaining a significantly low error rate of only 1.67%. Furthermore, this groundbreaking system couples high performance with substantial efficiency. Its low complexity enables real-time operation on standard central processors, capable of generating point cloud representations of the environment.

In conclusion, the proposed methodology makes it possible to design a point cloud segmentation algorithm aimed at efficiently and accurately segmenting curb and downward-facing stair edges in real-time scenarios within the real world. The system proved highly performing and reliable. It is also highly scalable and versatile in its nature.

- How can the reliability and robustness of the proposed systems be rigorously evaluated under diverse and unpredictable real-world conditions, taking into consideration indoor and outdoor scenarios in certain cases?

To comprehensively assess the reliability and robustness of the algorithm presented in Chapter 4, a rigorous, multi-stage stress test was conducted in diverse real-time scenarios around the University of Kent's Canterbury campus. The evaluation protocol encompassed two distinct phases. Initially, the object detection neural network's

performance was meticulously tested using a curated image dataset containing a wide variety of pothole examples. Subsequently, the system was evaluated using a publicly available, real-time video recording, enabling analysis of the network's performance under dynamic conditions. During these initial phases, key metrics including accuracy, frame rate, and detection rate were carefully calculated.

The system was then deployed in the field, undergoing trials at various times of day, under diverse weather conditions, and while operating at different speeds. Throughout these tests, 105 cases were meticulously documented, with detection confidence and relevant measurements recorded for each instance making sure that the ground truth is obtained with a strict and rigorous measurement method, that is ensured via using a tape measure. Analysis of the resulting dataset yielded the average accuracy, average error rate, and overall detection rate, providing valuable insights into the system's performance. The proposed system demonstrated salient reliability and versatility across these challenging conditions, validating the potential of this innovative technique for real-world implementation.

To comprehensively assess the system presented in Chapter 5, it underwent a rigorous two-stage evaluation process. Firstly, 67 outdoor test instances were conducted across the university campus. The system was deployed to detect and assess the characteristics of existing wheelchair and bicycle ramps. During these field tests, the system's calculated inclination angle, pre-inclination angle, and width measurements were meticulously recorded. To ensure the validity of these calculations, ground truth data was obtained using a tape measure and a digital inclinometer.

The second stage of the evaluation took place within a controlled environment. Here, the system was subjected to 16 carefully designed test cases, with each case representing a precise, one-degree increment in the inclination angle. In every instance, ground truth measurements were meticulously established. Width measurements were verified using a tape measure, while inclination accuracy was rigorously validated through the use of both a digital inclinometer and a protractor for cross-referencing. Throughout both the field and controlled experiments, average detection accuracy and error rate were meticulously documented. These metrics provide compelling evidence to assert the system's robust performance and unwavering reliability.

To rigorously evaluate the algorithm proposed in Chapter 6, 60 test cases across diverse indoor and outdoor environments were conducted. These tests were designed to expose the system to a wide range of curbs and downward-facing stairs, meticulously assessing its reliability. Ground truth measurements were carefully obtained using a tape measure, providing an accurate benchmark to validate the system's performance. Individual test results were then subjected to thorough analysis, with a focus on the precision of the system's measurements. This comprehensive evaluation confirmed the system's reliability and demonstrated its successful achievement of the intended design goals.

In conclusion, achieving a rigorous and comprehensive evaluation of a system's reliability and robustness necessitates meticulous testing procedures that encompass a broad spectrum of real-world conditions. Employing a multitude of carefully designed test cases is paramount. These test cases should ideally represent the vast array of potential scenarios the system may encounter in actual deployment. This comprehensive approach ensures that the system's performance is not merely optimised for a limited set of circumstances but rather demonstrates a consistent ability to function effectively under diverse and often unpredictable real-world constraints. It is important to acknowledge that certain systems may necessitate additional testing within controlled, isolated environments to assess their capabilities and limitations fully.

- How can the proposed systems be utilised for the purpose of providing an additional layer of safety to traditional mobility-focused assistive technologies? What benefit do they introduce?

The systems presented in this thesis demonstrate salient reliability and scalability. Their adaptability enables seamless integration with a wide range of assistive technologies (ATs), particularly those designed to enhance mobility. These systems have the ability to provide real-time, spatially accurate representations of the surrounding environment, offering crucial insights for safe navigation. They effectively detect hazards such as negative road anomalies, inclines (both upwards and downwards), curbs, ramps, drops,

and stairs, mitigating the risk of falls and injuries, whether the assistive device relies on autonomous, semi-autonomous, or manual control.

By utilising these innovative systems, users of mobility-related ATs gain a newfound level of freedom and independence. No longer constrained by the limitations that once restricted their movement and safety, they can navigate their surroundings with greater confidence. These systems represent a significant step towards breaking down barriers caused by mobility impairments, promoting inclusivity and enabling individuals to engage more fully in society. Moreover, by making autonomous navigation more achievable, these systems liberate AT users from the need to focus constantly on mobility limitations, allowing them to dedicate their attention to what truly matters.

In conclusion, the systems introduced in this thesis introduce an additional layer of safety to mobility-related assistive technologies. They empower individuals by making safe navigation a priority, significantly reducing the potential for falls and injuries. Furthermore, these systems pave the path towards both autonomous and semi-autonomous navigation solutions in the field of assistive technologies. As a result, users of mobility-related ATs are granted greater freedom of movement, enabling them to participate more actively in the world around them.

7.3 Future Work Recommendations

This research presents several avenues for further development and refinement, including:

Expanding the Negative Road Anomaly Dataset: The manually annotated dataset can be significantly expanded to encompass a wider range of pothole examples, including those captured under extremely low-light conditions. Furthermore, incorporating additional classes of negative road anomalies, such as cracks, drops, ravelling, rutting, edge breaks, and corrugations, would greatly enhance the system's versatility and robustness.

Depth Assessment and Traversal Analysis: A critical future research goal is to augment the negative road anomaly detection system by incorporating depth estimation

capabilities. This will empower the system to determine whether a detected obstacle is traversable in a worst-case scenario where avoidance strategies are not feasible. Careful consideration must be given to the trade-offs between additional sensing modalities, accuracy improvements, computational complexity, and real-time performance requirements.

Enhancing the Inclination Detection System: Exploring the integration of LiDAR technology holds the potential to improve the accuracy of the inclination detection system significantly. A key focus of this exploration will be robust segmentation and refined width estimation techniques, specifically addressing performance under ultra-low light conditions.

Optimising Traversal Area Detection: To further enhance the performance of the curb and downward-facing stairs avoidance system, future research will investigate the use of Time-of-Flight (ToF) sensor arrays. This approach promises to enhance measurement accuracy and increase fault tolerance, leading to a more reliable and robust system.

Providing a detailed assessment of the impact of vibrations on the proposed systems: Although vibration mitigation was included in the design and implementation of all three proposed systems, the impact of vibration on the performance of the systems was not assessed. Future work will include an in-depth analysis of the impact of vibration on all three systems including an assessment of the systems' behaviour with and without the vibration mitigation process.

References

[1] "AI based socially assistive robotics," UK Research and Innovation - EPSRC, 2018. [Online]. Available: https://gtr.ukri.org/projects?ref=studentship-2112938. [Accessed 10 12 2020].

[2] Y. Kim, B. Velamala, Y. Choi, Y. Kim, H. Kim, N. Kulkarni and E.-J. Lee, "A Literature Review on the Smart Wheelchair Systems," *ArXiv,* 2023.

[3] B. Paden, M. Cáp, S. Z. Yong, D. Yershov and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles," *arXiv,* 2016.

[4] A. A. H. Yahya Tawil, "Deep Learning Obstacle Detection and Avoidance for Powered Wheelchair," in *Innovations in Intelligent Systems and Applications Conference (ASYU)*, 2022.

[5] World Health Organisation, Wheelchair Provision Guidelines, 2023.

[6] A. Sohail, M. A. Cheema, M. E. Ali, A. N. Toosi and H. A. Rakha, "Data-driven approaches for road safety: A comprehensive systematic literature review," vol. 158, no. 105949, 2023.

[7] B. Erickson, M. Hosseini, Mudhar and P. e. al, "The dynamics of electric powered wheelchair sideways tips and falls: experimental and computational analysis of impact forces and injury.," 2016.

[8] L. Jensen, User perspectives on assistive technology: a qualitative analysis of 55 letters from citizens applying for assistive technology, vol. 69, 2014, pp. 42-45.

[9] L. Fehr, W. Edwin Langbein and S. B. Skaar, Adequacy of power wheelchair control interfaces for persons with severe disabilities: A clinical survey, vol. 37, 2000, pp. 353-360.

[10] Y. Matsumotot, T. Ino and T. Ogsawara, Development of intelligent wheelchair system with face and gaze based interface, IEEE, pp. 262-267.

[11] George Constantin Rascanu and Razvan Solea, Electric wheelchair control for people with locomotor disabilities using eye movements, Sinaia, Romania: IEEE, 2011.

[12] P. Arora, A. Sharma, A. S. Soni and A. Garg, Control of wheelchair dummy for differently abled patients via iris movement using image processing in MATLAB, IEEE, 2015, pp. 1-4.

[13] M. Henderson, S. Kelly, R. Horne, M. Gillham, M. Pepper and J.-M. Capron, Powered Wheelchair Platform for Assistive Technology Development, IEEE, 2014, pp. 52-56.

[14] S. Chatzidimitriadis, S. M. Bafti and K. Sirlantzis, Non-Intrusive Head Movement Control for Powered Wheelchairs: A Vision-Based Approach, vol. 11, 2023, pp. 65663-65674.

[15] P. Viswanathan, J. Little, A. K. Mackworth and A. Mihailidis, Adaptive Navigation Assistance for Visually-Impaired Wheelchair Users, 2011.

[16] J. Leaman and H. M. La, A Comprehensive Review of Smart Wheelchairs: Past, Present, and Future, vol. 47, 2017, pp. 486-499.

[17] S. K. Sahoo and B. B. Choudhury, Design and implementation of advanced sensor systems for smart robotic wheelchairs: A review, vol. 14, 2024, pp. 151-168.

[18] L. Matthies and A. Rankin, Negative obstacle detection by thermal signature, IEEE, pp. 906-913.

[19] B. César-Tondreau, G. Warnell, K. Kochersberger and N. R. Waytowich, Towards Fully Autonomous Negative Obstacle Traversal via Imitation Learning Based Control, vol. 11, 2022, p. 67.

[20] F. Chollet, Deep Learning with Python, Manning, 2017.

[21] V. Pereira, S. Tamura, S. Hayamizu and H. Fukai, A Deep Learning-Based Approach for Road Pothole Detection in Timor Leste, IEEE, 2018, pp. 279-284.

[22] S. Anand, S. Gupta, V. Darbari and S. Kohli, Crack-pot: Autonomous Road Crack and Pothole Detection, vol. abs/1810.05107, 2018.

[23] A. Kendall, V. Badrinarayanan and R. Cipolla, Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder, vol. abs/1511.02680, 2015.

[24] V. Badrinarayanan, A. Kendall and R. Cipolla, SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, vol. 39, 2017, pp. 2481-2495.

[25] V. Badrinarayanan, A. Handa and R. Cipolla, SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust, vol. abs/1505.07293, 2015.

[26] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally and K. Keutzer, SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 1MB, vol. abs/1602.07360, 2016.

[27] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, vol. 25, F. Pereira, C. J. Burges, L. Bottou and K. Q. Weinberger, Eds., Curran Associates, Inc., 2012.

[28] L. Zhang, F. Yang, Y. Daniel Zhang and Y. J. Zhu, Road crack detection using deep convolutional neural network, IEEE, 2016, pp. 3708-3712.

[29] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary and A. Agrawal, Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection, vol. 157, Elsevier, 2017, pp. 322-330.

[30] K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, 2015.

[31] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li and Li Fei-Fei, ImageNet: A large-scale hierarchical image database, IEEE, 2009, pp. 248-255.

[32] P. Cunningham and S. J. Delany, k-Nearest Neighbour Classifiers - A Tutorial, vol. 54, 2022, pp. 1-25.

[33] L. Breiman, Random Forests, vol. 45, 2001, pp. 5-32.

[34] P. Geurts, D. Ernst and L. Wehenkel, Extremely randomized trees, vol. 63, 2006, pp. 3-42.

[35] E. Theodoros, Support Vector Machines: Theory and Applications, V. S. C. D. Paliouras Georgios and Karkaletsis, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 249-257.

[36] Deeplearning.stanford.edu, Unsupervised Feature Learning and Deep Learning Tutorial., 2019.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and É. Duchesnay, Scikit-learn: Machine Learning in Python, vol. 12, 2011, pp. 2825-2830.

[38] L. K. Suong and J. Kwon, Detection of Potholes Using a Deep Convolutional Neural Network, vol. 24, 2018, pp. 1244-1257.

[39] J. Redmon and A. Farhadi, YOLO9000: Better, Faster, Stronger, 2016.

[40] C.-W. Kuan, W.-H. Chen and Y.-C. Lin, Pothole Detection and Avoidance via Deep Learning on Edge Devices, IEEE, 2020, pp. 1-6.

[41] H. N. Srikanth, D. S. Reddy, D. K. Sonkar, R. Kumar and P. Rajalakshmi, Pothole Detection for Autonomous Vehicles in Indian Scenarios using Deep Learning, IEEE, 2023, pp. 184-189.

[42] F. Zhang, H. Peng and L. Yu, SIP-YOLOv5: A Road Negative Obstacle Detection Network Based on Improved YOLOv5 and Prediction Box Correction Algorithm, IEEE, 2022, pp. 468-473.

[43] A. Chitholian, "Pothole object detection dataset," 2020. [Online]. Available: https://public.roboflow.com/object-detection/pothole. [Accessed 24 01 2024].

[44] B. A. Studio, "Potholes Dataset," 2020. [Online]. Available: https://aistudio.baidu.com/projectdetail/4355010. [Accessed 01 02 2024].

[45] E. Moscoso Thompson, A. Ranieri, S. Biasotti, M. Chicchon, I. Sipiran, M.-K. Pham, T.-L. Nguyen-Ho, H.-D. Nguyen and M.-T. Tran, SHREC 2022: Pothole and crack detection in the road pavement using images and RGB-D data, vol. 107, 2022, pp. 161-171.

[46] Aparna, Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal and A. Akula, Convolutional neural networks based potholes detection using thermal imaging, vol. 34, 2022, pp. 578-588.

[47] K. He, X. Zhang, S. Ren and J. Sun, Deep Residual Learning for Image Recognition, 2015.

[48] FLIRONE, "FLIR ONE Gen 3 Thermal Camera for Smart Phones | FLIR Systems," 2019. [Online]. Available: https://www.flir.co.uk/products/flir-one-gen-3/. [Accessed 28 05 2018].

[49] X. Yu and E. Salari, Pavement pothole detection and severity measurement using laser imaging, IEEE, 2011, pp. 1-5.

[50] N. Otsu, A Threshold Selection Method from Gray-Level Histograms, vol. 9, 1979, pp. 62-66.

[51] K. Azhar, F. Murtaza, M. H. Yousaf and H. A. Habib, Computer vision based detection and localization of potholes in asphalt pavement images, IEEE, 2016, pp. 1-5.

[52] Jianbo Shi and J. Malik, Normalized cuts and image segmentation, vol. 22, 2000, pp. 888-905.

[53] C. Koch and I. Brilakis, Pothole detection in asphalt pavement images, vol. 25, 2011, pp. 507-515.

[54] A. Fitzgibbon, M. Pilu and R. Fisher, Direct least squares fitting of ellipses, IEEE, 1996, pp. 253-257 vol.1.

[55] S.-K. Ryu, T. Kim and Y.-R. Kim, Image-Based Pothole Detection System for ITS Service and Road Management System, vol. 2015, 2015, pp. 1-10.

[56] I. Schiopu, J. P. Saarinen, L. Kettunen and I. Tabus, Pothole detection and tracking in car video sequence, IEEE, 2016, pp. 701-706.

[57] D. Ai, G. Jiang, L. Siew Kei and C. Li, Automatic Pixel-Level Pavement Crack Detection Using Information of Multi-Scale Neighborhoods, vol. 6, 2018, pp. 24452-24463.

[58] Y. Shi, L. Cui, Z. Qi, F. Meng and Z. Chen, Automatic Road Crack Detection Using Random Structured Forests, vol. 17, 2016, pp. 3434-3445.

[59] He Youquan, Wang Jian, Q. Hanxing, W. Zhang and X. Jianfang, A research of pavement potholes detection based on three-dimensional projection transformation, IEEE, 2011, pp. 1805-1808.

[60] Z. Zhang, X. Ai and N. Dahnoun, Efficient disparity calculation based on stereo vision with ground, IEEE, 2013, pp. 1-5.

[61] Z. Zhang, X. Ai, C. K. Chan and N. Dahnoun, An efficient algorithm for pothole detection using stereo vision, IEEE, 2014, pp. 564-568.

[62] G. H. Golub and U. von Matt, Quadratically constrained least squares and quadratic problems, vol. 59, 1991, pp. 561-580.

[63] X. Ai, Y. Gao, J. Rarity and N. Dahnoun, Obstacle detection using U-disparity on quadratic road surfaces, IEEE, 2013, pp. 1352-1357.

[64] A. Mikhailiuk and N. Dahnoun, Real-time pothole detection on TMS320C6678 DSP, IEEE, 2016, pp. 123-128.

[65] Y. Li, C. Papachristou and D. Weyer, Road Pothole Detection System Based on Stereo Vision, IEEE, 2018, pp. 292-297.

[66] U. Sinha, “Labelling connected components - Example,” 2018. [Online]. Available: : http://aishack.in/tutorials/labelling-connectedcomponents-example/.. [Accessed 20 08 2020].

[67] I. Moazzam, K. Kamal, S. Mathavan, S. Usman and M. Rahman, Metrology and visualization of potholes using the microsoft kinect sensor, IEEE, 2013, pp. 1284-1291.

[68] R. A. El-laithy, J. Huang and M. Yeh, Study on the use of Microsoft Kinect for robotics applications, IEEE, 2012, pp. 1280-1288.

[69] D. S. O. Correa, D. F. Sciotti, M. G. Prado, D. O. Sales, D. F. Wolf and F. S. Osorio, Mobile Robots Navigation in Indoor Environments Using Kinect Sensor, IEEE, 2012, pp. 36-41.

[70] P. Rakprayoon, M. Ruchanurucks and A. Coundoul, Kinect-based obstacle detection for manipulator, IEEE, 2011, pp. 68-73.

[71] D. A. Casas Avellaneda and J. F. López-Parra, Detection and localization of potholes in roadways using smartphones, vol. 83, 2016, pp. 156-162.

[72] W. Buttlar and S. Islam, "Integration of Smart-Phone-Based Pavement Roughness Data Collection Tool With Asset Management System," Islam2014. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/38287.

[73] Roadroid, " Road conditioning monitoring using smart phones. Quick Start Version 1.2.1," 2013. [Online]. Available: www.roadroid.com.. [Accessed 20 08 2020].

[74] C. Chellaswamy, H. Famitha, T. Anusuya and S. B. Amirthavarshini, IoT Based Humps and Pothole Detection on Roads and Information Sharing, IEEE, 2018, pp. 084-090.

[75] A. Mednis, G. Strazdins, M. Liepins, A. Gordjusins and L. Selavo, RoadMic: Road Surface Monitoring Using Vehicular Sensor Networks with Microphones, 2010, pp. 417-429.

[76] M. Wang, R. Birken and S. Shahini Shamsabadi, Implementation of a multi-modal mobile sensor system for surface and subsurface assessment of roadways, K. J. Peters, Ed., 2015, p. 943607.

[77] W. Kongrattanaprasert, H. Nomura, T. Kamakura and K. Ueda, Detection of Road Surface States from Tire Noise Using Neural Network Analysis, vol. 130, 2010, pp. 920-925.

[78] I. Abdić, L. Fridman, E. Marchi, D. E. Brown, W. Angell, B. Reimer and B. Schuller, Detecting Road Surface Wetness from Audio: A Deep Learning Approach, 2015.

[79] Q. Qiu, T. Yang and J. Han, A new real-time algorithm for off-road terrain estimation using laser data, vol. 52, 2009, pp. 1658-1667.

[80] N. Heckman, J.-F. Lalonde, N. Vandapel and M. Hebert, Potential negative obstacle detection by occlusion labeling, IEEE, 2007, pp. 2168-2173.

[81] B. Zheng, J. Shen, L. He, T. Hu, T. Cao and M. Li, A Method for Slope Detection of Planetary Surface in The Autonomous Obstacle Avoidance of Planetary Rover, IEEE, 2021, pp. 386-390.

[82] X. Meng, Z. Cao, L. Zhang, S. Wang and C. Zhou, A slope detection method based on 3D LiDAR suitable for quadruped robots, IEEE, 2016, pp. 1398-1402.

[83] R. B. Rusu, A. Sundaresan, B. Morisset, M. Agrawal and M. Beetz, Leaving Flatland: Realtime 3D Stereo Semantic Reconstruction, 2008, pp. 921-932.

[84] J.-S. Gutmann, M. Fukuchi and M. Fujita, 3D Perception and Environment Map Generation for Humanoid Robot Navigation, vol. 27, 2008, pp. 1117-1134.

[85] S. Singh, R. Simmons, T. Smith, A. Stentz, V. Verma, A. Yahja and K. Schwehr, Recent progress in local and global traversability for planetary rovers, IEEE, pp. 1194-1200.

[86] A. Murarka and B. Kuipers, A stereo vision based mapping algorithm for detecting inclines, drop-offs, and obstacles for safe local navigation, IEEE, 2009, pp. 1646-1653.

[87] A. Murarka, M. Sridharan and B. Kuipers, Detecting obstacles and drop-offs using stereo and motion cues for safe local motion, IEEE, 2008, pp. 702-708.

[88] S. Brossette, J. Vaillant, F. Keith, A. Escande and A. Kheddar, Point-cloud multi-contact planning for humanoids: Preliminary results, 2013.

[89] S. Cockrell, G. Lee and W. Newman, Determining navigability of terrain using point cloud data, 2013.

[90] C. K. Tseng, I. H. Li, Y. H. Chien, M. C. Chen and W. Y. Wang, Autonomous stair detection and climbing systems for a tracked robot, 2013.

[91] M. Nejati and B. D. Argall, Automated incline detection for assistive powered wheelchairs, IEEE, 2016, pp. 1007-1012.

[92] J. Wu, W. Hu, J. Coelho, P. Nitu, H. R. Paul, P. Madiraju, R. O. Smith and S. I. Ahamed, Identifying Buildings with Ramp Entrances Using Convolutional Neural Networks, IEEE, 2019, pp. 74-79.

[93] B.-F. Wu, Y.-S. Chen, C.-W. Huang and P.-J. Chang, An Uphill Safety Controller With Deep Learning-Based Ramp Detection for Intelligent Wheelchairs, vol. 6, 2018, pp. 28356-28371.

[94] C. Lutz, F. Atmanspacher, A. Hornung and M. Bennewitz, NAO walking down a ramp autonomously, IEEE, 2012, pp. 5169-5170.

[95] S. A. K. Tareen and H. M. Khan, Novel slope detection and calculation techniques for mobile robots, IEEE, 2016, pp. 158-163.

[96] Jin-Xia Yu, Zi-Xing Cai and Zhuo-Hua Duan, Dead reckoning of mobile robot in complex terrain based on proprioceptive sensors, IEEE, 2008, pp. 1930-1935.

[97] Zhibin Li, N. G. Tsagarakis and D. G. Caldwell, Stabilizing humanoids on slopes using terrain inclination estimation, IEEE, 2013, pp. 4124-4129.

[98] Y. Ashraf, A. Abdallah, A. Osman, E. E. D. Nehad and M. Fanni, Autonomous Detection of Stair Dimensions for the Motion Planning of Stair Climbing Robots, 2021.

[99] J. L. Apellániz, M. García, N. Aranjuelo, J. Barandiarán and M. Nieto, LiDAR-based curb detection for ground truth annotation in automated driving validation, IEEE, 2023, pp. 5054-5059.

[100] G. Zhao and J. Yuan, Curb detection and tracking using 3D-LIDAR scanner, IEEE, 2012, pp. 437-440.

[101] A. Y. Hata and D. F. Wolf, Feature Detection for Vehicle Localization in Urban Environments Using a Multilayer LIDAR, vol. 17, 2016, pp. 420-429.

[102] Y. Zhang, J. Wang, X. Wang, C. Li and L. Wang, A real-time curb detection and tracking method for UGVs by using a 3D-LIDAR sensor, IEEE, 2015, pp. 1020-1025.

[103] Y. Zhang, J. Wang, X. Wang and J. M. Dolan, Road-Segmentation-Based Curb Detection Method for Self-Driving via a 3D-LiDAR Sensor, vol. 19, 2018, pp. 3981-3991.

[104] J. Huang, P. K. Choudhury, S. Yin and L. Zhu, Real-Time Road Curb and Lane Detection for Autonomous Driving Using LiDAR Point Clouds, vol. 9, 2021, pp. 144940-144951.

[105] E. Horváth, C. Pozna and M. Unger, Real-Time LIDAR-Based Urban Road and Sidewalk Detection for Autonomous Vehicles, vol. 22, 2021, p. 194.

[106] W. Yao, Z. Deng and L. Zhou, Road curb detection using 3D lidar and integral laser points for intelligent vehicles, IEEE, 2012, pp. 100-105.

[107] Z. Yu, H. Zhu, L. Lin and H. Yang, Robust Curb Detection Based on the Accessible Route Analysis and Key Frames Prediction, IEEE, 2021, pp. 24-29.

[108] G. Wang, J. Wu, R. He and S. Yang, A Point Cloud-Based Robust Road Curb Detection and Tracking Method, vol. 7, 2019, pp. 24611-24625.

[109] D. Guo, G. Yang, B. Qi and C. Wang, Curb Detection and Compensation Method for Autonomous Driving via a 3-D-LiDAR Sensor, vol. 22, 2022, pp. 19500-19512.

[110] Y. Zhu, D. Han, B. Xue, J. Jiao, Z. Zou, M. Liu and R. Fan, Road Curb Detection Using A Novel Tensor Voting Algorithm, IEEE, 2019, pp. 590-595.

[111] Y. Jung, S.-W. Seo and S.-W. Kim, Curb Detection and Tracking in Low-Resolution 3D Point Clouds Based on Optimization Framework, vol. 21, 2020, pp. 3893-3908.

[112] K. Yamamoto, K. Kanai and J. Katto, Curb Height Estimation using 2DLiDAR mounted on e-Bike, IEEE, 2020, pp. 389-390.

[113] J. Guerrero, R. Chapuis, R. Aufrere, L. Malaterre and F. Marmoiton, Road Curb Detection using Traversable Ground Segmentation: Application to Autonomous Shuttle Vehicle Navigation, IEEE, 2020, pp. 266-272.

[114] M. Zou and Y. Kageyama, Road Curb Detection Based on a Deep Learning Framework, IEEE, 2023, pp. 0259-0262.

[115] U. Patil, A. Gujarathi, A. Kulkarni, A. Jain, L. Malke, R. Tekade, K. Paigwar and P. Chaturvedi, Deep Learning Based Stair Detection and Statistical Image Filtering for Autonomous Stair Climbing, 2019.

[116] H. Zhou, H. Wang, H. Zhang and K. Hasith, LaCNet: Real-time End-to-End Arbitrary-shaped Lane and Curb Detection with Instance Segmentation Network, IEEE, 2020, pp. 184-189.

[117] E. Mihankhah, A. Kalantari, E. Aboosaeedan, H. D. Taghirad and S. A. A. Moosavian, Autonomous staircase detection and stair climbing for a tracked mobile robot using fuzzy controller, 2009.

[118] Jaemin Byun, Junyoung Sung, Myung Chan Roh and Sung Hoon Kim, Autonomous driving through Curb detection and tracking, IEEE, 2011, pp. 273-277.

[119] E. Pollard, J. Perez and F. Nashashibi, Step and curb detection for autonomous vehicles with an algebraic derivative-based approach applied on laser rangefinder data, IEEE, 2013, pp. 684-689.

[120] M. P. Manuel, K. Balan, M. Murad and M. Krishnan, A Novel Curb Detection Method for Advanced Driver Assistance Systems, IEEE, 2018, pp. 360-363.

[121] Y. Cai, K. Ou, D. Li, Y. Zhang, X. Zhou and X. Mou, Robust Real-Time Curb Detection for Autonomous Sanitation Vehicles, IEEE, 2023, pp. 475-481.

[122] S. Panev, F. Vicente, F. De la Torre and V. Prinet, Road Curb Detection and Localization With Monocular Forward-View Vehicle Camera, vol. 20, 2019, pp. 3568-3584.

[123] M. Cheng, Y. Zhang, Y. Su, J. M. Alvarez and H. Kong, Curb Detection for Road and Sidewalk Detection, vol. 67, 2018, pp. 10330-10342.

[124] J. Rhee and J. Seo, Low-Cost Curb Detection and Localization System Using Multiple Ultrasonic Sensors, vol. 19, 2019, p. 1389.

[125] S. A. Bouhamed, I. K. Kallel and D. S. Masmoudi, Stair case detection and recognition using ultrasonic signal, 2013.

[126] J. Razavi and T. Shinta, A novel method of detecting stairs for the blind, IEEE, 2017, pp. 1-5.

[127] S. Hatua, H. Bose and D. N. Ray, Dynamic and Optimized Model for Stairs Detection, 2020.

[128] Tzutalin, "LabelImg. Git code (2015).," [Online]. Available: https://github.com/tzutalin/labelImg. [Accessed 15 01 2024].

[129] J. Dib, K. Sirlantzis and G. Howells, "An Annotated Water-Filled, and Dry Potholes Dataset for Deep Learning Applications," *Mendeley Data,* vol. V1.

[130] J. Dib, K. Sirlantzis and G. Howell, “An annotated water-filled, and dry potholes dataset for deep learning applications,” *Data in Brief,* vol. Volume 48, no. 109206, 2023.

[131] J. Redmon, “Darknet: Open Source Neural Networks In C.,” 2020. [Online]. Available: Available at: <https://pjreddie.com/darknet/>. [Accessed 1 July 2022].

[132] J. Dib, K. Sirlantzis and G. Howells, “A Review on Negative Road Anomaly Detection Methods,” *IEEE Access,* vol. 8, pp. pp.57298-57316, 2020.

[133] “RAC Pothole Index – statistics and data for UK roads,” RAC.co.uk, 10 November 2023. [Online]. Available: https://www.rac.co.uk/drive/advice/driving-advice/rac-pothole-index-statistics-data-and-projections/. [Accessed 27 December 2023].

[134] “"Pothole pandemic:" Drivers made 63 compensation claims a day for pothole damage last year,” Liberal Democrats, 26 December 2023. [Online]. Available: https://www.libdems.org.uk/press/release/pothole-pandemic-drivers-made-63-compensation-claims-a-day-for-pothole-damage-last-year#:~:text=Pothole%20compensation%20claims%20have%20risen,1.66m%20the%20previous%20year.. [Accessed 27 December 2023].

[135] G. Huang, Z. Liu, L. v. d. Maaten and K. Q. Weinberge, “Densely connected convolutional networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (pp. 2480-2490)*, Las Vegas, NV, USA, 2016.

[136] K. He, X. Zhang, S. Ren and J. Sun, “Deep residual learning for image recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778)*, Boston, MA, USA, 2015.

[137] A. Bochkovskiy, C.-Y. Wang and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” *Computing Research Repository (CoRR),* vol. abs/2004.10934, 2020.

[138] Intel Corporation, *Intel RealSense Product Family D400 Series Datasheet,* 2023.

[139] BBC News, “Rod Stewart & friends fix potholes in road, fed up with inaction (UK) - 9th March 2022,” 2022.

[140] National Health Service - UK, “National Wheelchair Data Collection Quarterly Publication Files 2023-24,” 2023-24. [Online]. Available: https://www.england.nhs.uk/statistics/statistical-work-areas/national-wheelchair/national-wheelchair-data-collection-quarterly-publication-files-2023-24/. [Accessed 27 01 2024].

[141] S. Sonenblum, S. Sprigle and J. Martin, "Everyday sitting behavior of full-time wheelchair users," *Journal of Rehabilitation Research and Development,* vol. 53, no. 5.

[142] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.

[143] SmartToolFactory, "SmartToolFactory - Github," [Online]. [Accessed 06 02 2024].

[144] Consumer Product Safety Commission, "National Electronic Injury Surveillance System 2003-2022," NEISS Online Database, April, 2023.

[145] H. Xiang, A.-M. Chany and G. Smith, "Wheelchair related injuries treated in US emergency departments', Injury Prevention," *Injury Prevention - BMJ,* pp. 8-11, February 2006.

Appendix A

Negative Road Anomalies Detection Techniques - Additional Information

A.1 Vision-Based Systems

A.1.1 Deep Learning-Based Techniques

Visible-Light RGB Camera as Input:

Pereira et al. [21] training process employed the following parameters:

- Number of epochs: 200
- Number of images in the training set: 13,244
- Number of images in the validation set: 3,250
- Batch size: 16
- Optimiser: Adam Optimiser (cost-function reduction method)
- Learning Rate: 0.0001
- β1=0 .9
- β2=0 .999
- epsilon= $1e^{-8}$
- Cost-function: $C = -yi \log ai - (-1 - yi)\log (1 - ai)$

Over-fitting avoidance: 20% dropout (random dropping of neurons when training)

Pereira et al. demonstrated promising results. However, their approach exhibits several limitations requiring further development:

- **Real-time Applicability:** The method lacks real-time detection capabilities, operating solely on still images. This hinders its use in dynamic traffic scenarios.
- **Dataset Transparency:** The lack of a publicly shared dataset and limited visual examples in the publication impede reproducibility and raise questions regarding the training and testing data's representativeness.

- **Environmental Robustness:** The absence of testing under low-light or high-brightness conditions raises concerns about the system's reliability in diverse real-world environments.
- **Safety and Redundancy:** Sole reliance on machine vision for pothole detection poses potential risks in real-time scenarios due to the susceptibility of imaging systems to environmental factors. The lack of failsafe mechanisms warrants further consideration.

Anand et al [22] achieved their training process within the below parameters:

- Number of epochs: 20
- Batch size: 64
- Optimiser: Adam Optimiser (cost-function reduction method)
- Learning Rate: 0.00001
- Codewords K = 32
- Momentum: 0.9
- Training data: The authors have used 2 different sets in order to test their system:
    - GAPs dataset: Image Size 1920x1080 pixels
        - Number of images in the training set: 1,418
        - Number of images in the testing and validation set: 551
    - Zhang dataset: Image Size 3264x2448 pixels [28]
        - Number of images in the training set: 1.3 million
        - Number of images in the testing and validation set: 0.7 million

By training the network using both datasets, the achieved results are:

GAPs Dataset:

- Accuracy: 99.893%
- F1-score: 72.14%

ICIP Dataset:

- Accuracy: 92.37%
- F1-score: 93.01%

The method proposed by Anand et al. demonstrates promising experimental results. However, several limitations warrant careful consideration for real-world implementation:

- **Environmental Robustness:** The method's performance under low-light or extreme brightness conditions remains untested. Such variations could significantly degrade image quality, hindering accurate detection.

- **Contextual Factors:** The experimental setting (time of day, weather, location) is unspecified. These factors can profoundly influence image characteristics and system reliability.
- **Texture-Based Vulnerability:** The system's reliance on texture creates susceptibility to false positives from surfaces with similar textural qualities to potholes.
- **Water-Filled Potholes:** The system may fail to detect potholes filled with opaque water, as this would obscure the characteristic texture.

Gopalakrishenan et al [29] employed a training process that involves:

- **Preprocessing:** Pavement images are preprocessed, likely through cropping or downsampling, to reduce dimensions and remove extraneous edges.
- **Labelling:** Images are manually labelled to establish ground truth for supervised learning.
- **Transfer Learning:** The pre-trained VGG-16 model's weights are fine-tuned on the pavement image dataset, adapting its learned features to the crack detection task.
- **Classifier Evaluation:** The fine-tuned VGG-16 is used as a feature extractor, with its outputs fed into various classifiers for performance comparison. These classifiers include Single Nearest Neighbor [32], Random Forest [33], Extremely Randomised Trees [34], Support Vector Machines [35], and Logistic Regression [36].

Training took place with the parameters below:

- For the Single NN:
    - Image Size: 224x224 pixels
    - Number of Neurons in Hidden Layer: 256
    - Dropout value: 0.5
    - Hidden Layer Activation: ReLU
    - Output Layer Activation: softmax
    - Image Batch Size: 32
    - Number of Epochs: 50
- All the other classifiers: 'scikit-learn' machine library in Python [37] was used with its out-of-the-box parameters.

Gopalakrishnan et al. report promising experimental results using their proposed method. However, several limitations warrant consideration:

- **Environmental Robustness:** The system's performance under challenging lighting conditions (low-light or extreme brightness) and in the presence of water-filled or reflective cracks remains untested. These factors could significantly degrade image quality and hinder accurate detection.
- **Real-World Validation:** The presented results are based solely on controlled dataset samples. Evaluation in real-world scenarios is essential to assess the system's robustness to real-time variations and potential failure modes.
- **Robustness to Occlusions and Diverse Textures:** The system's ability to reliably detect potholes obscured by water, debris, or irregular textures remains untested. This raises concerns about its robustness in real-world scenarios, where such conditions are likely to be encountered.

Suong et al [38] introduced modifications to the baseline YOLO architecture, resulting in their custom-designed variant. These modifications include:

- **Layer Elimination:** Removal of layers 23, 24, and 29, resulting in a substantial reduction of approximately 30 million parameter computations.
- **Filter Expansion:** Incorporation of a 2048-sized filter into layer 23. This was realised by increasing the filter count of the 26th convolutional layer from 64 to 256.
- **Depth Reorganisation:** Restructuring of layer 24's depth to 13 x 13 x 1024, consequently modifying layer 25's depth to 1024.
- **Routing Layer Integration:** Routing of layers 25 and 26 with the 22nd convolutional layer.
- **Anchor Box Adaptation:** Adjustment of the baseline model's anchor box dimensions (width and height) to establish anchors specific to the novel architecture.

The authors enhanced their proposed YOLO architecture with a self-implemented "den-anchor" component, synthesised by integrating elements from denser grid and anchor box models. To train their networks, the authors compiled a dataset reflecting diverse conditions and employed the following parameters:

- Existing YoLo Architecture:
    - 996 training images containing 1796 potholes and 203 testing images.
    - Learning rate: 1e-5 (from 0 to 100 epochs) and then 1e-6 (from 100 to 200 epochs)
    - Retrained for another 300 epochs
- Proprietary architecture:
    - 996 training images containing 1796 potholes and 203 testing images.
    - Learning rate: 1e-5 (from 0 to 100 epochs) and then 1e-6 (from 200 to 600 epochs)

- Trained for another 100 epochs using the Den-anchor

Despite promising experimental results, the work of Suone et al. exhibits significant limitations warranting consideration:

- **Lack of Real-Time Evaluation:** The authors fail to assess their system under real-time constraints. Their results are derived from offline analysis, precluding conclusions about real-world performance.
- **Inherent Vulnerability of Vision-Reliant Systems:** The exclusive reliance on machine vision for pothole detection raises concerns for real-time deployment. Environmental factors (e.g., lighting, weather) can degrade image quality, potentially compromising the system's reliability and posing safety risks to users who lack redundant safeguards.
- **Insufficient Precision for Safe Application:** The reported precision of approximately 82.5% is inadequate for reliable navigation. This level of accuracy leaves substantial room for misclassification, which could have dangerous consequences in a safety-critical system.

Thermal Imaging as Input:

Aparna et al.'s [46] proprietary CNN was configured as follows:

- Train-validation split: 90:10
- Image size: 240 × 295
- Total categories: 2.
- Total images: 4904
- Training dataset size: 4320
- Validation dataset size: 480
- Test dataset size: 104
- Kernel: 3 × 3 for convolution layers
- Activation: ReLU for convolution layers
- Loss function: binary cross entropy

As for ResNet, 3 different ratios were used during training and validation:

- 60:40
- 80:20

- 90:10

An attempt by using different ResNet models was also tested:

- ResNet18
- ResNet34
- ResNet50
- ResNet101
- ResNet152

Publically-available results are as follows::

Proprietary Network:

- Average training accuracy: 55.74%
- Average validation accuracy: 68.99%
- Training and Validation losses on still higher side
- Test accuracy: 73.06%

ResNet:

- ResNet18: Best accuracy 90.52% and validation loss of 27.37%
- ResNet34: Best accuracy 89.42% and validation loss of 27.57%
- ResNet50: Best accuracy 91.77% and validation loss of 24.07%
- ResNet101: Best accuracy 92.50% and validation loss of 22.40%
- ResNet152: Best accuracy 91.66% and validation loss of 22.28%

To further optimise the system, the the top-performing ResNet architectures (ResNet50, 101, and 152) were selected and re-evaluated using an 80:20 training/validation split and experimented with image sizes of 224x224 and 240x240 pixels. Larger image sizes encountered out-of-memory errors, highlighting potential hardware limitations. This refined testing resulted in improved accuracy, with "almost similar" performance observed between the two tested image sizes.

The achieved results were as follows:

Proprietary Model:

- Average Training Accuracy: 62.63%
- Average Validation Accuracy: 69.8%

ResNet:

- Average Training Accuracy: 94.64%
- Average Validation Accuracy: 95.52%

Several significant findings were reported:

- **ResNet Superiority:** ResNet-based CNN architectures outperformed the properietary custom-built model. ResNet50 and ResNet101 demonstrated particularly strong performance, suggesting their suitability for this pothole detection task.
- **Image Dimensions:** An input image size of 224x224 pixels appears optimal for ResNet models in this context.
- **Training Refinement:** Employing cyclic learning rates led to a noticeable improvement in model accuracy.
- **Low False Positives:** The system exhibited a low false-positive rate, indicating a promising level of reliability.

Limitations

- **Insufficient Specificity:** All test cases were grouped, obscuring the system's performance specifically with wet or reflective potholes. This raises concerns about robustness under diverse real-world conditions.
- **Environmental Sensitivity:** Thermal camera images can be adversely affected by weather conditions, potentially leading to degraded input and compromised system performance.

Laser Imaging as Input:

Yu et. al's [49] method has been tested using a set of 100 images which includes ten examples of each distress. The results were not clear as they are simply represented with a table comparing three samples (2 potholes and one crack) and showing that the severity level / crack type were the same between the manual assessment and the method proposed.

The proposed system presents several critical limitations, raising concerns about its applicability and performance in real-world road conditions:

- Inadequate Evidentiary Basis: The authors fail to provide sufficient data to substantiate claims about the system's effectiveness, hindering a comprehensive performance assessment.
- Absence of Essential Metrics: The lack of reported false-positive and false-negative rates obscures the system's reliability. These metrics are crucial for quantifying the potential for incorrect distress identification, which has significant implications for road maintenance decisions.
- Unexplored Environmental Constraints: The reliance on laser imaging introduces inherent limitations. The system's ability to detect water-filled potholes, a common real-world scenario, remains untested and is likely compromised due to optical interference.
- Limited Training Dataset: The use of only 100 images for neural network training raises significant concerns about overfitting and the model's ability to generalise to the diverse range of pothole manifestations encountered in practice.
- Disregard for Real-World Variability: The system neglects the inherent unevenness of road surfaces. This oversight renders it susceptible to

substantial errors induced by irregular surfaces and image instability, factors not reflected in the presented results.

- Lack of Transparency: The authors' omission of information regarding the allocation of images between training and testing sets undermines the ability to evaluate the work's methodological rigor and the validity of the reported results.

A.1.2 Non Deep Learning-Based Techniques

Visible-Light RGB Camera as Input:

Azhar et. al's [51] the system follows the steps set out below:

- Grayscale Conversion: Images are initially converted from the RGB color space to grayscale.
- Resizing & Normalisation: Images are resized to a uniform 200 × 200 pixel dimension. Orientation is normalised between 0 and 180 degrees.
- Cell Division: The image is divided into a grid of 625 non-overlapping cells (25 × 25 pixels).
- Block Formation: Each 8 × 8 pixel cell is further subdivided into four smaller 4 × 4 pixel blocks.

The resultant vector with the help of the HOG technique is of size 1 × 20,000 (625 × 4 × 8). This vector fulfils the image classification task with the help of the Naïve Bayes classifier labelling the image with the help of the "maximum posterior probability" technique.

The system then employs the "normalised cuts" image segmentation technique, introduced by Shi and Malik [52], to partition the image. This technique groups visually

similar regions by analysing global image features, measuring the overall similarity and dissimilarity between potential groupings. Following segmentation, the image is divided into 12 distinct regions. Regions with a mean intensity threshold below 80 are classified as potholes, suggesting that potholes generally exhibit darker pixel values compared to intact pavement.

In addition to its suboptimal reliability, the system exhibits the following critical weaknesses:

- Computational Overhead: The system's intensive computational requirements preclude deployment on power-constrained, battery-operated platforms. The reported processing time (0.673 seconds for a 200x200 image) underscores the system's inability to scale to contemporary high-resolution images.
- High False-Negative Rate: A significant number of false negatives compromise the system's dependability for automated navigation. This indicates frequent failures in pothole detection, posing safety risks.
- Sensitivity to Illumination: The reliance on HOG features introduces inherent vulnerabilities to variable lighting conditions. Performance degradation in low light or overly bright scenarios is a well-documented limitation of this feature extraction technique.
- Shadow Misclassification: The technique's tendency to misclassify shadows as potholes, as noted in the paper, further diminishes its reliability. This highlights the need for more robust feature representations to distinguish between true potholes and shadow-induced artefacts.

Koch et al. [53]'s core methodology stages are as follows:

Segmentation:

- Noise Reduction: A 5x5 median filter mitigates noise.

- Histogram-based Thresholding: The "triangle algorithm" [37] dynamically determines a threshold based on histogram shape. Further refinement with a 1D median filter reduces histogram peak interference. Binarisation follows, separating foreground (potential potholes) from background.

Shape Analysis:

- Pothole Candidate Extraction: Linear artifacts and regions deemed too small are eliminated. Geometric properties of remaining regions (major axis length, centroid position, orientation angle) are used for preliminary pothole identification.
- Elliptical Shape Approximation: "Morphological thinning" reduces potential pothole regions to their skeletal form. Branching points are used to determine the major path, and elliptic regression [54] approximates the elliptical shape of the pothole's shadow.

Texture Comparison:

The results obtained were as follows:

- Filtering and Dilation: A sequence of filters prepares the image, followed by morphological dilation to mitigate filter artefacts.
- Feature Extraction: Texture features are extracted from both the candidate pothole region and its surrounding background.
- Classification: If the candidate region exhibits a coarser, grainier texture compared to the background, it is classified as a pothole.
- Dataset & Evaluation: The system was trained on 50 images and tested on 70 images. Thresholds were manually tuned to optimise performance.

Schiopu et al.'s [56] system works in several key stages:

- **Region of Interest (ROI) Selection:** To focus computational effort, the system first defines a Region of Interest (ROI) within each video frame. This ROI is determined offline using cues like estimated road edges (which converge at a vanishing point) and the maximum distance at which potholes are reliably visible.
- **Intensity-Based Candidate Generation:** Within the ROI, a thresholding algorithm is applied to identify pixels with unusually high-intensity values. These pixels form the initial set of potential pothole candidates.
- **Wayside Removal:** To eliminate roadside objects that might also be bright, the system calculates an intensity threshold and removes pixels below it, as these likely represent non-pothole elements.
- **Reflection Filtering:** Reflections from other vehicles or structures can create false positives. The system addresses this with an offline procedure that compares consecutive frames, analyses the mean intensity, and calculates a depth matrix to detect and remove reflections.
- **Shadow Discrimination:** Shadows, like potholes, can appear dark. The system distinguishes between them by analysing properties like shape regularity (shadows tend to follow object outlines), depth (estimated from the number of dark pixels), contour length and shape, and whether the candidate appears consistently over multiple frames.
- **Pothole Identification & Tracking:** Candidates that successfully pass all these filters are labelled as potholes. The system then tracks these potholes across consecutive frames, using car speed, camera parameters, and Euclidean distance calculations to estimate their movement and maintain a live record of their positions.

The presented method exhibits several limitations that warrant careful consideration, particularly for deployment in safety-critical applications:

- **Suboptimal Precision:** While a precision of 90% is encouraging, it may not be sufficient for a system responsible for real-time navigational decisions on moving platforms. The remaining 10% error rate could lead to misidentification of potholes, posing risks to users.
- **Speed-Dependent Tracking:** The reliance on estimated speed for tracking calculations introduces potential vulnerabilities. Accuracy may degrade at varying speeds, hindering the system's ability to reliably track pothole positions.
- **Vulnerability to Lighting Variation:** The method's performance under inconsistent lighting conditions remains untested. Sudden, intense light could significantly alter frame characteristics, potentially compromising the system's reliability.
- **Unexplored Nighttime Performance:** The lack of testing in nighttime conditions raises concerns about the system's ability to cope with a drastically different image profile. The abundance of dark pixels could pose significant challenges to the detection and tracking algorithms.

Dihao et. al.'s [57] proposed method was implemented in Python for evaluation and testing. The following parameters were used:

- Dataset: The CFD dataset [58] was employed, consisting of 118 images. This dataset presents challenges for crack detection algorithms due to the presence of noise artefacts (e.g., oil stains, watermarks, and variable lighting conditions).
- Data Allocation: The dataset was partitioned into a training set (70 images) and a testing set (48 images)

The presented method exhibits several limitations that warrant careful consideration, particularly for deployment in safety-critical applications:

- **Suboptimal Precision:** While a precision of 90% is encouraging, it may not be sufficient for a system responsible for real-time navigational decisions on moving platforms. The remaining 10% error rate could lead to misidentification of cracks, posing risks to users.
- **Computational Overhead:** The reliance on computationally intensive operations raises concerns about real-time viability and power demands. Further optimisation or alternative algorithmic approaches are needed to address this potential bottleneck.
- **Unexplored Robustness:** The lack of comprehensive testing under diverse conditions, including water-filled cracks and other image noise, obscures the method's ability to maintain performance in real-world scenarios.
- **Nighttime Performance:** The absence of nighttime testing leaves uncertainties about the system's ability to cope with drastically different image profiles where dark pixels predominate, presenting a significant challenge to crack detection.
- **Performance Metrics:** The omission of runtime data and resource usage analysis (e.g., CPU, memory) hinders a concrete assessment of the method's real-world suitability.

Stereo Vision as Input:

He Youquan et. al.'s [59]system exhibits significant weaknesses that compromise its practical utility:

- **Detection without Localisation:** The system detects potholes but lacks the ability to precisely determine their location.
- **Sensitivity to Lighting:** Results are highly dependent on the intensity of the LED light source and ambient lighting conditions, negatively impacting reliability in uncontrolled environments.
- **Hardware Dependence:** The system's accuracy is directly tied to CCD camera resolution and performance, introducing potential for variability.
- **Lack of Robustness:** The method provides no mitigation strategies for external light interference.
- **Insufficient Evaluation:** The paper offers limited performance data, with only a single discrepancy result.
- **Missing Context:** The paper omits crucial details about the testing environment (location, time, weather), hindering an assessment of the system's true capabilities.
- **Unknown Error Rate:** The absence of error rate data obscures the method's potential for misdetection.

Zhang et. al.'s [60] presented method (including its optimised version) exhibits several limitations that raise concerns about its robustness and performance in real-world scenarios:

- **Sensitivity to Lighting Conditions:** The system's reliance on visual input introduces susceptibility to errors induced by variable light intensity. Performance degradation in low-light or overly bright conditions is a significant drawback.
- **Challenges with Water or Ice:** The core detection principle likely fails when encountering water-filled or ice-covered potholes, as these disrupt the expected surface geometry upon which the method relies.
- **Insufficient Evaluation Data:** The lack of comprehensive results and accuracy metrics obscures the system's true reliability and error potential under diverse conditions.
- **Missing Environmental Context:** The omission of testing details (location, time, weather) hinders assessment of the system's adaptability to variable real-world environments.
- **Unknown Error Rate:** The absence of error rate data (false positives, false negatives) leaves critical questions about the system's reliability unanswered.

Li et. al's [65]method has had some noticeable weaknesses:

- The system's performance relies on two stereo cameras, which need to be calibrated every time the code runs.
- The system will fail when trying to detect water or ice-filled potholes as the cameras will not be able to produce a valid output.
- The system's performance will be severely affected by the light intensity as it relies solely on two RGB cameras.
- The paper did not provide any data which can be used to assess the success and failure rates of the system.
- The paper only mentions the location where the system is tested but does not contain any data which describes the time and weather conditions in which the system was tested.

Depth Camera as Input:

Moazzam et. al.'s [67] method is comprised of the steps below:

- Area Calculation: Depth images are binarised at millimetre increments. Pixel counts transformed to real-world units, yield area estimates at each depth level.
- Volume Approximation: An area vs. depth curve is generated, and the trapezoidal rule (unit spacing) is applied to approximate pothole volume.
- Visualisation: Contour plots represent depth slices, and 3D plots depict pothole geometry in real-world coordinates (mm).
- Classification: The system classifies potholes as "Squared Decay", "Longitudinal", or "Cube-like" based on area decay patterns and calculated geometric properties (centroid, eccentricity, orientation).

The proposed system includes some noteworthy limitations:

- **Sensitivity to Lighting:** The system's reliance on camera-based depth sensing introduces vulnerability to errors induced by variable light intensity (low light or overly bright conditions).
- **Challenges with Water or Ice:** Water-filled or ice-covered potholes disrupt the surface geometry upon which the method relies, likely leading to detection failures.
- **Error Rate:** The reported error rate of 15% raises concerns about the system's reliability in real-world applications.

A.2 Non-vision-Based Systems

Tire Pressure / Vibration as Input:

The weaknesses of such systems are as follows:

- **Post-Detection Limitation:** By detecting potholes only after impact, these systems offer no pre-emptive avoidance capability.

- **Confounding Factors:** Reliance on tyre pressure or vibrations introduces susceptibility to false positives triggered by uneven roads, curbs, and other non-pothole events.
- **Lack of Localisation:** Such systems typically provide no precise localisation information necessary for an autonomous vehicle to navigate around a pothole.
- **Environmental Vulnerability:** Performance is likely to be heavily influenced by external factors such as vehicle speed, tyre type, and road conditions.
- **Assessment Inapplicability:** Due to their reactive nature, these methods cannot fulfil the core requirement of pre-detection necessary for autonomous vehicle safety. Therefore, they are not suitable for assessment under your proposed criteria.

Appendix B

Incline Detection and Assessment Techniques - Additional Information

B.1 Vision or Radar-Based Systems

B.1.1 Point-Cloud-Based Techniques

Qiu et. al [79], and Heckman et. al's [80] system was tested in a simulation-based environment and in real-time scenarios. However, the provided results do not illustrate the system's performance in different scenarios or its capabilities for assessing the inclination angle of the detected plane.

The method seems promising; however, some major weaknesses are identified:

There is not enough evidence to illustrate the system's performance in angle estimation

Only one scenario was used to test the system in real-time, the rest is just simulation data

- **Environmental Sensitivity:** Laser scanning systems demonstrate susceptibility to bright sunlight, which compromises performance by diminishing the strength of the return signal.
- **Reflectivity-Induced Errors:** The presence of highly reflective surfaces can lead to false readings, undermining the accuracy of laser scanning measurements.
- **Geometric Limitations:** Targets with steep inclinations may prove challenging to detect with laser scanning systems, highlighting the influence of target geometry on system efficacy.
- **Safety Constraints:** The deployment of accurate laser rangefinders is often restricted due to ocular safety concerns, with many models exceeding safe exposure limits, precluding their use in public areas.

Zhend et. al's [81] system limitations are present:

- **Reliance on LiDAR and Laser Sensing:** The system's exclusive use of LiDAR and laser sensing introduces vulnerabilities to false and erroneous readings, particularly when encountering reflective surfaces (e.g., water-covered inclined planes) or ultra-bright environments.
- **Power Consumption:** The combined use of LiDAR and laser sensing raises concerns about high power consumption, potentially limiting operational time or requiring substantial power infrastructure.
- **Real-time Performance Uncertainty:** While real-time performance is suggested, insufficient data prevents a conclusive assessment. Rigorous testing under diverse conditions is needed to verify real-time capabilities.

Meng et al.'s [82] experimental findings provided reveal that the estimated slope angles closely align with the actual values, with acceptable variances, highlighting the method's rotational invariance to some extent. This rotational invariance is crucial for ensuring consistent and reliable slope detection across different observation orientations. The small errors in angle estimation and low variance in estimated angles further underscore the method's effectiveness in accurately detecting slopes using 3D LiDAR data, showcasing its potential for enhancing the capabilities of quadruped robots in navigating complex environments.

Although the system demonstrated reliable results, important limitations still exist:

Reflectiveness limitations: LiDAR sensing techniques, as mentioned previously, are prone to many false positives and false negatives in scenarios involving reflective surfaces and water-covered surfaces.

Bright Sunlight: Intense ambient light can overwhelm the returning laser signal, making it difficult to distinguish the target, especially at long distances.

Atmospheric Interference: Weather conditions such as fog, rain, snow, and dust can scatter or absorb the laser pulses, reducing accuracy, range, or even preventing measurements entirely.

Power-consumption: LiDAR is known to require high-power consumption which could affect the performance level required of the system.

Rusu et al. [83], Gutmann et. al. [84], Snigh et al. [85], and Murarka et. al's [86, 87] method employed a segmentation algorithm the 3D grid that is crucial for identifying distinct ground regions that may be potentially traversable by the robot. The algorithm considers voxel columns within a planning radius around the robot's current pose, focusing on horizontal or slightly inclined surfaces suitable for wheeled mobile robots. Surfaces with high inclines are deemed non-traversable, and segments are defined as collections of voxel columns with the same height, representing level and inclined ground regions. Non-segmented voxel columns are considered obstacles or unreachable areas, aiding in creating a local safety map for path planning.

The local safety map generated by the algorithm annotates cells with labels such as Level, Inclined, Non-ground, or Unknown, based on the safety and traversability of the corresponding regions. Level and Inclined cells are considered safe for navigation, while Non-ground cells indicate obstacles or unsafe areas. Safe cells can further be annotated as Potential Drop-off Edges if a drop-off is detected, providing detailed information for path planning. By utilising stereo vision as the primary sensor, the algorithm offers a cost-effective and information-rich alternative to laser sensors, enabling comprehensive hazard detection in urban environments.

The system provides promising results, however, certain weaknesses have been identified:

- **Binary Solution:** The described approach only addresses the problem in a binary technique. This means that the exact inclination angle, and width of the plane are not extensively monitored and accurately measured.
- **Disparity Matching:** Stereo vision requires computationally expensive algorithms (e.g., Semi-Global Matching) to find corresponding pixels between the left and right images, which can strain embedded systems or necessitate specialised hardware acceleration.

- **Real-time performance limitations:** The use of the occupancy grid techniques introduces significant limitations and latency to the system as it could, on many occasions, cause significant delays and require a high computational power.
- **Error-rate and fault - tolerance:** Although the system produces a decent accuracy, the error rate is higher than or almost equal to 5% whether it is with False Positives or False Ngeatives.

Brossette et al. [88], Cockrell et al. [89], and Tseng et. al. [90], and Nejati et al.'s [91] algorithm consists of two main steps: identifying inclined planes in the vicinity of the robot and determining if these planes are navigable. The algorithm segments the candidate ramps via the region-growing algorithm and then detects traversable ramps by finding the convex hull and defining the four corners of the incline. Additionally, the algorithm provides information on the location, orientation, dimensions, incline amount, and length of the ramp, enabling the wheelchair to safely navigate the identified ramp.

Even though the described algorithm returns promising results, some weaknesses remain as its Achille's Heel:

- **Limited Real-World Applicability:** An average accuracy of 87.9% raises concerns about the system's reliability in practical applications. For deployment in safety-critical or precision-dependent scenarios, a segmentation algorithm should demonstrate a significantly higher degree of accuracy to ensure the fidelity of the resulting segmentation masks.
- **Non-Uniform Point Density:** Large variations in point density can make it challenging to establish consistent and reliable similarity criteria across the point cloud.
- **Overlapping Objects:** Region growing may struggle to distinguish closely spaced objects or objects with subtle boundaries, leading to merging of distinct regions which could affect the width estimation of the plane.
- **Computational Power:** The system's reliance on a region-growing algorithm for candidate plane segmentation raises concerns about its computational efficiency. The inherent iterative nature of region-growing techniques can introduce significant computational overhead, potentially jeopardising the system's ability to perform reliably in real-time scenarios where timely processing is critical

B.1.2 Non-Point-Cloud-Based Techniques

J. Wu et. al's [92] methodology used in the research project involved building an artificial intelligent system using Convolutional Neural Networks (CNN) to classify building images based on whether they have a handicap ramp or not. The researchers recruited volunteers, mainly graduate students, to collect images of building entrances from urban areas like Milwaukee, Chicago, and Los Angeles. The images were then used to train the CNN model for image classification of building accessibility. The CNN model consisted of convolution layers and pooling layers, which scanned each pixel of the image to extract main features. These features were then sent to the fully connected layer where the probabilities of each class (accessible or inaccessible) were calculated based on the extracted features

Additionally, the Wu et al. followed a structured approach for model construction:

- Specifying the input shape for images in the first layer.
- Adding layers with hyperparameters using functions like add() provided by Sequential().
- Compiling the model by specifying the optimiser, loss function, and metrics.
- Fitting the neural network with the training dataset, epochs, and batch size for training purposes

Furthermore, the research incorporated the use of Dropout function in the CNN model to reduce overfitting. This function helps in avoiding overfitting by randomly removing neurons during training and making the model less sensitive to specific neurons, thus improving generalisation and model performance.

Although the system's accuracy was high (95.6%), many limitations hinder the system's overall performance:

- **Specificity and Safety:** The binary classification approach fails to address crucial geometric parameters (inclination, width) essential for distinguishing wheelchair ramps from potentially hazardous alternatives like bicycle ramps. This oversight poses a significant safety risk.
- **Environmental Robustness:** Exclusive reliance on an RGB camera introduces vulnerability to low-light conditions, potentially leading to failures in poorly lit environments. This constraint significantly restricts real-world usability.

- **Computational Demands:** The system's utilisation of a CNN necessitates a powerful GPU-based processor. This introduces constraints in terms of power consumption and hardware requirements, hindering practical deployment in mobile or resource-limited settings.
- **Real-Time Performance:** The observed latency in the system's operation excludes it from use cases requiring real-time decision-making for safe and effective navigation.

B. Wu et. al's [93] system also addresses the L-type narrow ramp scenario, where the system adjusts the speed to ensure safety while turning corners. The gyroscope data is used to estimate the wheelchair's inclination on the ramp. Experimental results show that the controller with ramp detection significantly outperforms the system without ramp detection in terms of travel time and energy consumption.

The system employs a deep learning-based ramp detection model, which is trained using the CNN-4 structure and the Q-learning-based ANFIS controller. The ramp detection model demonstrates high accuracy in classifying ramps of different appearances, while the controller calculates safe speeds based on the ramp classification results. The controller is able to prevent overturning and optimise the riding experience by considering environmental information and wheelchair feedback.

Although the system presents a significant improvement over other systems, it still falls below the expected performance due to the reasons below:

- **Input Sensor Limitations:** The system relies on an RGB sensor as an input source, this means that the system's performance will be significantly hindered in low-light, and limited visibility scenarios.
- **Classification Limitations:** The system only relies on RGB data and assesses inclination post-traversal via the use of the gyroscope. This means that the system is prone to many errors such as considering bicycle ramps as wheelchair ramps until the wheelchair actually accesses the ramp putting the user under the risk of significant harm.
- **Computational Requirements:** The system utilises a CNN with a fuzzy logic-based decision network; this means that a highly-performing processor is required; hence, the system could not be utilised in real-life scenarios.

Lutz et al.'s [94] method addresses challenges such as foot slippage, inaccurate hardware calibration, and joint backlash affecting the robot's motion, the study emphasises the need for the robot to correct its heading regularly while walking down the

ramp. The robot's heading is corrected based on IMU data to prevent collisions with the handrail or falls due to changes in yaw resulting from the ramp's inclination. Statically stable steps are executed on the ramp, and heading corrections are made to ensure safe navigation. The method also includes learning basic motions for the Nao robot to navigate ramps effectively. Kinesthetic teaching is applied to teach the robot single-stepping motions, optimising them for static stability on the ramp. By manually moving the robot to statically stable poses and recording keyframes, smooth movements are created using Bezier curves. The learned motions include entering and exiting the ramp, walking steps on the ramp, and small correction steps to account for drift and correct the robot's orientation.

This technique although successful, has many limitations such as:

- **Ambiguous Scale:** A single camera image lacks direct cues about the absolute distance to objects. This can lead to misinterpretation of object sizes and distances, complicating navigation tasks and collision avoidance.
- **Difficulty with Stationary Objects:** Without motion cues (e.g., parallax) provided by stereo or multi-view setups, monocular systems may struggle to reliably detect and estimate the distance to stationary objects.
- **Lack of important criteria:** The proposed system does not take into consideration the inclination of a plane until the robot traverses it so that the inclination is picked up by the IMU. This means that the robot could be exposed to a bicycle ramp and it will attempt to traverse it putting the user at risk of falls.

B.2 Non-Vision or Radar-Based Systems

Tareen et al.'s [95] technique, although can be achieved at a low cost, is prone to many limitations such as:

- **Temperature and Humidity:** The speed of sound, on which ultrasound sensors rely, is affected by the temperature and humidity of the air. Changes in these conditions can lead to measurement inaccuracies.
- **Air Turbulence:** Wind and air currents can distort or deflect the sound waves, leading to unreliable readings and making it harder to track objects accurately.

- **Susceptibility to Noise:** Background noise, especially at frequencies similar to the sensor's operating frequency, can interfere with the emitted signals and make it difficult for the sensor to detect its intended targets.
- **Soft or Angled Surfaces:** Soft materials (like fabric) tend to absorb sound waves rather than reflect them. Angled surfaces may deflect sound waves away from the sensor. This makes reliable detection very difficult.

Yu et al.'s [96] technique is prone to many limitations due to the following:

- **Vibration Sensitivity:** Accelerometers cannot distinguish true tilt changes from the vibrations. This makes them unreliable on moving platforms.
- **Drift:** Over time, accelerometers can suffer from drift in their readings, leading to growing inaccuracies in incline calculations.
- **Need for Initial Reference**: Gyroscopes don't give absolute incline. They can only detect changes from the starting point. Without a good initial reference point, the incline estimates will become inaccurate.
- **Sensitivity to Magnetic Interference:** If included, magnetometers (which sense magnetic fields) can improve accuracy, but they are highly susceptible to interference from nearby metal objects or magnetic fields.

Zhiblin et al.'s [97] proposed technique is prone to many limitations, as per below:

- **Surface Irregularities:** Real-world surfaces aren't perfectly smooth. Bumps, textures, and small obstacles can all skew contact sensor readings, leading to inaccurate incline estimations.
- **Contact Point Variation:** The exact point of contact between the sensor and the surface can shift slightly. This variability affects the measurements and thus the calculated incline.
- **Sensor Wear:** Over time, contact sensors can wear down due to friction and abrasion, leading to a gradual decline in their accuracy.
- **Calibration and Maintenance:** The mechanical components used in contact exploration systems may need regular calibration and maintenance to ensure accuracy.
- **Vibrations and Movement:** Unexpected vibrations or movements of either the robot or the environment can disrupt the contact and impact measurement reliability.

Appendix C

Downwards-facing Stairs and Curbs Detection and Localisation Techniques

C.1 Vision or Radar-Based Systems

C.1.1 Point-Cloud-Based Techniques

Ashraf et al. [98] presented the main framework of the proposed approach, including steps such as clustering data based on the normal vector and utilising the Pinhole Camera Model to compute the point cloud.

Although the proposed approach achieved a good level of accuracy, some drawbacks have been identified:

- **Scope:** The proposed approach exclusively addresses the detection of upwards-facing stairs, neglecting the complexities inherent in detecting downwards-facing stairs. Downwards-facing stairs often present distinct challenges due to varying perspectives and potential for occlusions.
- **Experimental Rigor:** The system's efficacy has been evaluated solely within a simulation environment. To comprehensively gauge the system's capabilities, it is crucial to conduct rigorous testing in real-world scenarios with their inherent complexities.
- **Computational Efficiency:** Simulation results indicate that the system necessitates substantial computational resources. This potential limitation underscores the need for optimisation or alternative approaches to facilitate real-time applications.
- **Field of View Constraints:** The intrinsic design of pinhole cameras results in a limited field of view. This characteristic might hinder the system's suitability for applications demanding comprehensive situational awareness.
- **Low-Light Performance:** The restricted aperture of pinhole cameras can compromise performance in low-light environments, potentially leading to image noise and a loss of detail critical for accurate stair detection.
- **Occlusion Sensitivity:** The reliance of stereo vision on establishing correspondences between images renders the system vulnerable to occlusions.

Obstructions can impede the matching process, compromising the accuracy of the generated depth map.

Apellániz et al. [99] also covered related works in curb detection using LiDAR data and the methodology's experimental results, demonstrating a significant reduction in manual annotation time. The process involves annotating curbs in a 3D point cloud reconstruction map. Two datasets are introduced, the first one with imprecise estimations in line format and the second one with annotations in point format. A DNN training and testing process using a DeepLabv3+ model is described for identifying and separating different curb instances. The dataset distribution for training, validation, and testing is outlined with approximately 80% for training, 15% for validation, and 5% for testing.

Although the system exhibits a high accuracy rate, it is prone to some limitations:

- **Precision Constraints:** The system's maximum precision of 87.8% renders it unsuitable for deployment in critical assistive technologies, where exceptionally high reliability is paramount for user safety.
- **Functionality:** The system only performs the detection task. It lacks the localisation task that enables it to determine the safe traversable area accurately.
- **Data Density and Processing:** LiDAR generates large point clouds. Processing these vast datasets in real-time for curb detection can be computationally demanding, requiring powerful hardware.
- **Sensitivity to Weather:** Adverse weather conditions like fog, heavy rain, or snow can affect LiDAR's performance. These conditions can scatter or absorb the laser beams, leading to less accurate or incomplete data.
- **False Positives from Small Objects:** LiDAR can sometimes struggle to distinguish between curbs and other similar-height objects along the roadside, such as low walls, raised planters, or debris. This can lead to false positives in curb detection.
- **Cost Implications:** LiDAR sensors can be significantly more expensive than traditional camera-based systems. This cost factor might be a barrier to widespread adoption in certain applications.

Zhao et al. [100], Hata et al. [101], Zhang et al. [102, 103], Huang et al. [104], Horváth et al. [105] , Yao et. al. [106], Yu et al. [107], Wang et al. [108], Guo et al. [109] , Zhu et al. [110], Jung et al [111], Yamamoto et al. [112], and Gurrero et al. [113] technique's limitations are as follows:

- **Inadequate Performance:** The system performance (83.34% accuracy) falls below the minimum accuracy expectations for it to be considered usable for assistive technologies.
- **Weather Limitations:** LiDAR, as mentioned in previous sections, is prone to many false and inaccurate readings when the weather conditions are challenging due to the nature of the sensing technique.
- **Data Density and Processing Power:** LiDAR returns highly dense point clouds, which means that a high computational power is required making it unsuitable for real-time detection on assistive technologies.
- **Cost Implications:** LiDAR is one of the most expensive sensors and utilising it would bring new cost implications to the system's cost.

C.1.2 Non-Point-Cloud-Based Techniques

Zou et al.'s [114] approach involves training and evaluating the feature extractor models, selecting the best model for curb detection based on the Faster R-CNN framework, and conducting experiments to assess performance metrics such as accuracy, sensitivity, specificity, F1 score, precision, area under the curve (AUC), and prediction time. The results obtained illustrated that the customised CNN network outperforms the original ResNet18 network in terms of accuracy, sensitivity, F1-score, AUC, and prediction time. Additionally, the detection framework using the customised CNN network shows an improved average precision (AP) compared to the original ResNet18 detector. However, the presence of noise in input images affects the detection results, suggesting the need for pre-processing images in the future to enhance performance.

Although the system's accuracy and precision reported are high, the system has many weaknesses:

- **Insufficient Empirical Validation:** Whilst the reported accuracy appears promising, the combination of low average confidence scores and numerous false positives as illustrated in the figure presented raises concerns regarding the representativeness of the reported metrics. To establish a robust assessment of the system's performance, a more comprehensive testing regime is necessary.
- **Functional Limitations:** The system only performs the detection task, it lacks the localisation capabilities.
- **Lack of Segmentation:** The system's reliance on manually specified bounding boxes for curb detection indicates a lack of object segmentation capabilities. This constraint hinders its ability to autonomously isolate the curb within the

image, even when the curb occupies a small portion of the provided bounding box.

- **Weather Limitations:** The sensory method used (RGB Camera) is highly susceptible to weather limitations as it is prone to failure in severe weather conditions, and low-light scenarios.
- **Processing Power Limitations:** The system detects curb via the use of a CNN, a method that requires high computational power, leading to more power consumption which limits the system's usability.

Zhou et al.'s [116] LaCNet utilises deep metric learning to map lane and curb pixels into an embedding space for accurate differentiation. The network includes components for class segmentation and pixel grouping to rebuild instances and distinguish between lanes and curbs. The overall architecture and loss functions of LaCNet are detailed, with an equal weighting distribution among the loss components. Experimental evaluation on different datasets shows the effectiveness of LaCNet in detecting random number and arbitrary-shaped lanes and curbs in real-time. The network performed well when exposed to arbitrary-shaped lanes and curbs irrespective of the vehicle's angle, achieving a processing speed of 32 FPS and more than 98% F1 measure on a self-collected dataset.

The system achieved a good accuracy at a real-time pace, however, some limitations render it not usable for assistive technologies:

- **Lighting:** Dramatic changes in lighting (harsh shadows, extreme brightness, nighttime) significantly alter image appearance, making it difficult for the CNN to generalize well.
- **Weather:** Rain, snow, fog, and glare can obscure curbs or create misleading visual cues, impacting detection accuracy.
- **Image Quality:** Poor focus, motion blur, or camera noise can degrade image quality, hindering the CNN's ability to extract meaningful features.
- **Textureless Curbs:** Curbs with minimal colour or texture variation provide fewer visual cues for the CNN to learn from, making detection more challenging.
- **Low or Gradual Curbs:** The visual distinction between road surfaces and very low or gradually sloping curbs might be too subtle for the CNN to reliably detect.
- **Power Consumption:** Although the system performs in real-time, it requires an advanced processor with a good graphical processor unit (GPU), this requires a high power consumption render it not mountable of platforms with limited battery power.

Mihankhah et al. [117] designed their system configuration with three laser sensors, the use of a first derivative of altitude for detecting steps and stairs, and the challenges of segmenting steps accurately. The approach involves calculating the first derivative of altitude for each data point obtained by the laser sensors. This also includes deriving the altitude based on the travelled distance over the scanning angle. The system also addresses the challenges posed by variable point density and noise in the data.

The system also includes a stair detection algorithm for pedestrians using laser sensors. It uses a recursive line fitting algorithm and sliding window length considerations for calculating the first derivative of altitude. The detection of peaks in the first derivative of altitude is explained along with the method to detect steps. The process involves peak detection in a noisy environment to identify changes in ground orientation. The system basically ensures peak detection and merging information from two sensors.

Although the technique seems promising, many limitations exist:

- **Oblique Angles:** Laser beams oriented at very shallow angles to the curb might not reflect enough energy back to the sensor, making detection difficult, especially for lower curbs.
- **Occlusions:** Obstructions like parked cars, vegetation, or pedestrians can block the laser beams and hinder curb detection.
- **Curb Types:** Laser range functions might struggle to reliably detect curbs with irregular shapes, very gradual slopes, or those made from materials with low reflectivity.
- **Weather:** Rain, fog, snow, or dust can scatter or absorb laser beams, significantly reducing the range and accuracy of the sensor.
- **Sunlight:** Bright ambient light can introduce noise and affect the accuracy of laser range measurements.
- **Sensor Cost:** High-resolution laser sensors, especially those with multiple beams or long ranges, can be significantly expensive. This can limit their use in budget-constrained applications.

Manuel et al. [120], Cai et al. [121] and Panev et al.'s [122] approach includes curb edge extraction, parameterised 3-D curb template fitting, and the use of support vector machine classifiers. The system was tested on a database of 11 videos, validated with LIDAR measurements and manual labels. The system uses computer vision techniques, machine learning, and temporal filtering for real-time accuracy and flexibility in detecting

curbs. The comparative analysis shows the effectiveness of this method compared to other techniques like LIDARs, ToF cameras, laser range finders, and stereo cameras in detecting curbs accurately in a variety of conditions.

Although the system achieved an acceptable average accuracy, some limitations prevent it from being chosen as a potential solution to be mounted onto assistive technologies for the reasons set out below:

- **Non-linear Distortion:** Fisheye lenses introduce significant barrel distortion, where straight lines appear curved. This complicates algorithms designed for traditional cameras and makes geometric reasoning about the scene more challenging.
- **Depth Distortion:** The non-uniform stretching of the image distorts the perception of depth and distances, making accurate curb localisation difficult.
- **Concentration of Pixels:** Fisheye lenses concentrate a large amount of the field of view into the centre of the image. This leads to significantly lower resolution at the image edges, where important curb details might be lost.
- **Complex Distortion Models:** Calibrating fisheye cameras requires more complex distortion models than standard pinhole cameras. Small calibration errors can significantly impact the accuracy of curb detection.
- **Computational Resource Constraints:** The system exhibits a substantial demand for computational power, necessitating a high-performance GPU. This requirement translates to high energy consumption, rendering it unsuitable for integration with assistive technologies that rely on limited-capacity batteries lacking regenerative braking capabilities.

Cheng et al.'s [123] system uses spatial features to refine the detection and eliminate false curb points found in areas like the border of grass. Curb-like areas are eliminated by considering their location relative to curbs. Vanishing point detection is described along with the creation of a cost map for the road detection using different types of information such as flatness, gradient, curb position, and path link, and how these contribute to finding optimal road borders. The method also ensures the construction of the weighted graph cost map and the importance of curb-induced cost in enhancing road detection performance. A voting mechanism is used to identify the vanishing point and calculate various costs for road detection such as flatness, gradient, curb, and link costs. The algorithm outlines steps for identifying the two base points with the smallest average cost paths and determines road borders. Following this, sidewalk detection is discussed,

wherein initial sidewalk points are chosen based on geometry features. A region-growing method is employed to identify the sidewalk region, with a refinement strategy to remove false points. The process includes setting height and geometry constraints, comparing points for selection, and analysing error margins for curb detection accuracy using the F1-measure.

Although the system provides a promising result, its weaknesses render it not suitable for assistive technologies:

- **Descriptor Specificity:** The chosen descriptor might not capture all variations in curb appearance (different materials, textures, or damaged curbs).
- **Ambiguity:** Other objects with similar geometric characteristics to curbs might lead to misclassifications.
- **Vanishing Point Accuracy:** Inaccurate vanishing point estimation, especially in complex urban scenes, will affect the reliability of the road model.
- **Planar Road Assumption:** The method may struggle in areas where the road surface is significantly non-planar.
- **Texture Variations:** Sidewalk textures can vary widely (concrete, brick, etc.). This might impact the seed selection and growth process within the region-growing method.
- **Seed Point Requirements:** Incorrect seed points, especially in complex scenes, can lead to inaccurate sidewalk segmentation.
- **Computational Constraints:** The system's reliance on computationally intensive segmentation and classification algorithms renders it unsuitable for deployment in assistive technologies with limited power resources.

C.2 Non-Vision or Radar-Based Systems

Rhee et al. [124], Bouhamed et al. [125], Razavi et al. [126], and Hatua et al. [127]'s system's modelling hierarchy consists of two phases: Comparative Study and Detection Approach. The Comparative Study phase involves proper height selection and length identification for constructing the basic structure of the algorithm. Real-time measurements are taken from twelve different staircases to analyse the average height and length. Proper height selection ensures sensors are placed correctly. The Detection Approach involves implementing an

automated stair detection algorithm based on mathematical criteria. The system is tested on stairs of differing dimensions, producing positive results.

Although the system exhibits acceptable result, it is limited due to the following:

- **Reflections from Multiple Surfaces:** Ultrasound waves can bounce off multiple surfaces in the environment (e.g., walls, handrails), leading to ambiguous distance measurements and potential misinterpretations.
- **Angle Dependence:** Such sensors are heavily dependent on the tilting angle. This means that curbs or stairs, especially those with curved or irregular treads, might not reflect the ultrasonic waves directly back to the sensor, leading to inaccurate distance readings.
- **Material Composition:** Stair materials that absorb sound (like carpets) can significantly reduce the effectiveness Temperature and Air Pressure: The speed of sound is affected by temperature and air pressure. Uncompensated environmental changes can lead to measurement errors.
- **Air Currents or Obstructions:** Air drafts or small objects within the sensor's range can disrupt the acoustic signal and skew the readings of ultrasonic sensors.