Terrain characterization methods of unstructured terrain for an autonomous mobile robot

Erik Skultety - candidate number 117

Report submitted as part the degree of
Master in Applied Computer and Information Technology (ACIT)
- Phase 3 ACIT5930
60 credits

Department of Computer Science
Faculty of Technology, Art and Design

OSLO METROPOLITAN UNIVERSITY

Autumn 2023

Terrain characterization methods of unstructured terrain for an autonomous mobile robot

Erik Skultety - candidate number 117

Terrain characterization methods of unstructured terrain for an autonomous mobile robot

http://www.oslomet.no/

Printed: Oslo Metropolitan University

Abstract

In this thesis, the feasibility of a deep-learning-based terrain characterization method was assessed in comparison to a traditional analytical approach. Both solutions were implemented on a wheeled mobile robot equipped with a standard stereo depth camera, an IMU, and a GPS unit. Classical technique was derived from existing literature, while a deep-learning based implementation was developed alongside the navigation system and data processing utilities. The study conducted extensive experiments in a real-world setting located in an unstructured forest environment and gathered results in the form of energy consumption and roughness. While both methods proved effective at navigation from point A to point B, inherent limitations highlighted avenues for future advancements. Key findings include the need for an extensive, high-resolution dataset to optimize machine learning performance and a more robust navigational with global perception. The outcomes of this research pave the way for future exploration into refining terrain characterization techniques for more diverse environments and applications.

Keywords: terrain characterization, deep-learning, wheeled mobile robot, autonomous navigation, traversability

Acknowledgments

This master's thesis has been a project which has been going on for over a year, and I would like to express my thanks to the people who've helped me throughout my master studies and the writing of this thesis. Firstly, I want to thank my supervisor, Associate Professor Tønnes Nygaard, for his insights into the field of robotics and helping me establish a theme for this paper. Secondly, I also want to thank my fellow students at ACIT for their help throughout the past two years. Thirdly, I want to thank OsloMet for funding the robotic platform for both my project and hopefully many other projects to come. Lastly, I want to thank Dafne for supporting me throughout both my bachelor's and master's degree.

Contents

Chapter 1

Introduction

1.1 Motivation

Ever since the Mars Exploration Rover ”Opportunity” got trapped in a sand dune on its mission’s sol 447 (Angelova et al., 2007), every subsequent rover sent to the surface of Mars has continually become more capable at navigating safely to increase our reach in planetary exploration. While needlessly wandering off into dangerous terrain is never the objective, many of the most desired scientific curiosities can be found in the most challenging areas (Li & Lewis, 2022). However, episodes such as the Fukushima Daiichi nuclear disaster have proven that our robotic capabilities are not yet mature enough for an effective disaster response when needed (Nagatani et al., 2013). Among these is the capability to navigate through surroundings that were unexpectedly altered or previously unseen, and avoid hazardous areas in order persevere towards the mission’s objective.

While a regular car will perform optimally on dry asphalt road, it would take more effort and an experienced driver to operate the same type of car on a dirt road. In a similar fashion, a mobile robot set to operate in an environment that is predictable and known, will generally be designed with that kind of environment in mind. In contrast to this, unstructured environments can consist of unexpected and previously unseen occurrences. This means that a robot will have to exhibit additional sensing and navigational capabilities to be successful. As autonomous robots are envisioned to operate in remote or inaccessible environments, their potential applications often demand uninterrupted functioning without any human intervention. Any setback for the robot in such situations could lead to mission failure or significant delays. Therefore, it becomes paramount for the robot to have the capability to navigate challenges, anticipate issues, and find solutions autonomously.

Mobile robot navigation is an active subfield within robotics that consistently lead to exciting results, such as pushing the boundaries of planetary exploration or increased safety and efficiency at manufacturing facilities (Robotnik, 2023). However, some mobile robot applications encounter higher difficulty imposed by their surrounding terrain than others, leading to the challenges previously mentioned. Terrain classification and characterization are both possible techniques that aim to solve that problem. The former gives a robot the ability to distinguish specific terrains that lie in front of them, such as grass, gravel or concrete. The latter expands on this concept by assessing a robot's ability to traverse any given part of a terrain with regards to its locomotion. In either case, a mobile robot that can autonomously reason and navigate across the terrain it needs to traverse can take on objectives that would otherwise not be feasible without extensive human intervention.

A range of other applications in field environments such as agriculture, surveying, defence, and search and rescue, indicate the potential for mobile robots with higher degree of navigational capabilities. A recent DARPA Grand Challenge aims to bridge the gap of the ability to travel across off-road terrain between autonomous vehicles and manned vehicles (DARPA, 2022). While indoor environments are not the focus of this project, unstructured terrain can still be present in certain scenarios, such as the one presented and challenged by ELROB (2022).

Finally, there are several ethical considerations that need to be taken into account when developing any robot. Often, the most discussed issue with regards to robotic solutions is that they could negatively affect the labour market. Current examples of such scenarios are manufacturing robots in factories, which have reduced the need for human labour. This is not fully applicable for this project, since mobile robots that are able to navigate unstructured terrain would generally be employed in inherently hazardous activities to aid humans. However, mobile robot possessing the required capabilities for those applications could also find use for military ends (Lin et al., 2011). Similar to autonomous drones used for military purposes, more capable ground vehicles could present another alternative for destruction, and give advantages to those with access to this technology. On the other hand, the mobile robots could also assist in peace operations such as explosive disposal, transporting wounded to safety, or reconnaissance of dangerous regions. Additionally, economical implications are a common point of contention with regards to planetary rovers. Technology used for planetary exploration is often considered too costly with insignificant benefits in the public's eyes (Lin et al., 2011). But in the context of mobile robots in unstructured terrain, the accomplishments within this criticized field can also be utilized in terrestrial applications.

1.2 Problem statement

The goal of the thesis following this report will be to explore various areas within the fusion of computer vision and robot navigation, specifically aimed towards autonomous operation of mobile robots in unstructured terrain. This will be achieved with map building and robot adaptation through visual-based terrain characterization. Previously developed solutions based on two distinct approaches will be adapted, adjusted and implemented on a wheeled mobile robot platform. This includes a traditional analytical method, and a machine learning method. The wheeled mobile platform will carry a suite of sensors, RGBD camera, inertial measurement unit (IMU) and a GPS unit, to provide information of its surroundings. Upon complete implementation, testing and comparison will be completed between both configurations in a number of real-world scenarios on unstructured terrain. Based on this process, this project will attempt to answer the following research question:

> **How do deep learning-based and classical analytical terrain characterization methods compare in real-world navigation for a wheeled mobile robot?**

To address the research question effectively, we have established the following constraints. These constraints not only ensure consistency and integrity in our experiments but also facilitate a focused examination of the potential research contributions.

1. **Environmental Constraints:**

   - *Forest terrain:* All empirical assessments and experiments will be conducted exclusively within an urban forest environment in the Viken region in Norway.
   - *Weather conditions:* Experiments are restricted to specific weather conditions, avoiding extremities such as heavy rain, snow, ice or mud.
   - *Lighting conditions:* Tests will be performed only during daylight hours and under consistent ambient light to minimize variations in visual sensor input.

2. **Operational Constraints:**

   - *Robot speed:* The robot will maintain a constant speed and turning radius throughout all tests to ensure that results are not influenced by velocity variations.
   - *Human intervention:* Once an experiment begins, no human intervention is allowed, except for the initial placement of the robot at the starting position.
   - *External software:* The robot operates solely based on the software loaded at the start of the experimentation phase.

3. **Hardware Constraints:**

   - *Fixed hardware configuration:* All experiments are conducted with a pre-defined hardware setup chosen at the onset of the project. No alterations or additions to this hardware setup will be made during the course of the experiments regardless of their viability.
   - *Onboard data processing:* All computational tasks and data processing required for navigation will be done onboard the robot, leveraging the initial hardware configuration.

These constraints are designed to establish a controlled environment for the experiments, ensuring that observed outcomes are primarily a result of the tested navigation techniques and not external variables.

Chapter 2

Background

The following chapter will present the main aspects of the relevant theory, as well as provide an overview over previous and related works. This is divided into three main sections that constitute the most important parts of realizing this master thesis project.

2.1 Wheeled mobile robots

Wheeled mobile robots, among the most common configurations in robotics, offer a balance of stability, simplicity, and efficiency. Their design and functionality have evolved over the years to cater to a variety of applications, from industrial automation to planetary exploration.

2.1.1 General overview

Mobile robots are robots with a locomotion system used for generating motion to traverse their surroundings. An important branch of this field are *legged* robots, that are inspired by effective biological approaches. Meanwhile, the class of mobile robots that employ *wheels*, *tracks* and other types of locomotion systems, are human inventions. Without the initial starting point provided by evolution, the latter categories of mobile robots currently exhibit a limited degree of mobility in extreme terrain (Li & Lewis, 2022). Li and Lewis (2022) argue that this is due to the current lower degree of understanding of motion generation in complex ground terrain compared to motion based on flight aerodynamics and underwater hydrodynamics. Since biologically inspired locomotion systems are more adept at overcoming challenging terrain, it can be applicable to replicate the same ideas that have developed in nature (Siegwart et al., 2011). A particular example would consist of an actively adaptable suspension in a similar fashion to legged mobile robots and their articulated limbs, which can lead to improvements regarding tip-over stability, reach and footprint (Iagnemma & Dubowsky, 2004). The difficulties of replicating biological systems into mechanical solutions were addressed by Siegwart et al. (2011).

Excluding sensing and navigation of a mobile robot, its general motion capabilities largely depend on the wheels and drive system used (Tzafestas, 2014a). In addition, holonomic constraints, or a lack thereof, will decide the degrees of freedom available to a system. Holonomic constraints are dictated by the dynamics of a mobile robot, and determine the amount of degrees of freedom available to a system.

The types of wheels used for mobile robots can be categorized as conventional and special wheels (Tzafestas, 2014a). On one end of the spectrum of the conventional wheels are the motor powered wheels that are often fixed in the forward and reverse direction with regards to the robot's reference frame. On the other end are the non-powered castor wheels that are free to rotate perpendicular to the wheel's rotational axis, often off-set from their attached joint. Identical rotation can be achieved by the powered wheels, but this will depend on the robot's drive system. Special wheels are designed to achieve additional range of motion in multiple directions, specifically used for omnidirectional mobile robots. These include universal, mecanum and ball wheels (Tzafestas, 2014a).

Rubio et al. (2019) describe that wheeled mobile robots can be classified by the drive system they employ: differential drive, car-type, omnidirectional and synchro drive. Understandably, each type of drive will be most suitable for certain applications, as mentioned by the author. With regards to mobile robots in unstructured terrain, the literature within this subfield almost exclusively utilizes differential drive and car-type wheeled mobile robots as their experimental platforms, in addition to tracked mobile robots.

Differential drive robots consist of at least two powered wheels situated on each side of a platform, fixed in the direction of motion. In the case of two-wheeled differential drive, a castor wheel is used for balance and stability (Tzafestas, 2014a). Wheels are independently powered and controlled to generate motion either along a circular curve, rotation around a fixed point, or simply forwards or backwards. The type of motion depends on the wheel speed and driven direction of each side of a robot. For example, differing speeds in the same direction result in motion along a circular curve, while same speed in differing directions result in stationary rotation. Skid steering is a special implementation of the differential drive for tracked mobile robots, and differs only in the advantages and disadvantages related to the use of tracks as opposed to wheels (Tzafestas, 2014a).

Car-type mobile robots, also referred to as *Ackerman steering*, is the standard steering utilized in cars. A minimum of one pair of steered wheels allows for rotation along a minimum radius, as this drive system cannot turn while stationary. There are several combinations of how many pairs of wheels are steered or not steered, which affects the rate of turn of a vehicle. The main design concept of Ackerman steering is that the rotational axes of every wheel meet in a common cross point while turning, in an effort to avoid geometrically caused wheel slippage. (Tzafestas, 2014a).

Examples of robotic platforms used in research that also conducted experiments in real-world environments can be seen in figure 2.1. The most common type in the reviewed literature is a four-wheel differential drive and tracked skid-steering mobile robots for terrestrial applications, and six-wheel steered platforms with a rocker-bogie suspension (commonly used for Martian rovers) for planetary applications.

2.1.2 Non-holonomic constraints

Non-holonomic constraints inherently restrict the motion possibilities of wheeled mobile robots. These limitations arise primarily due to the no-slip condition at the wheel-ground interface and the fixed orientation of the wheels relative to the robot's chassis (Kolmanovsky & McClamroch, 1995). Specifically, while wheels are designed to roll forward without slipping, they commonly resist motion perpendicular to their axis of rotation, preventing all sideways movement that isn't caused by slipping. Additionally, the fixed orientation of the wheels means that without changing direction or trajectory, lateral motion is impossible. Such constraints pose significant challenges for path planning and adaptive navigation in autonomous robots. Depending on a robot's drive system, movement can be realized despite these constraints.

(a)

(b)

(c)

(d)

(e)

Figure 2.1: (a) Pioneer 2-AT (Ojeda et al., 2006) (b) Husky (IEEE, 2011) (c) Pluto (Helmick et al., 2009) (d) PackBot (Tzafestas, 2014a) (e) maXXII (Galati & Reina, 2019)

2.1.3 Mobile robot navigation

The navigation of a mobile robot from one place to the other in any environment consists of a number of tasks in order to realize an autonomous movement. This includes sensing, map building and map interpretation, self-localization, path planning and motion control (Siegwart et al., 2011).

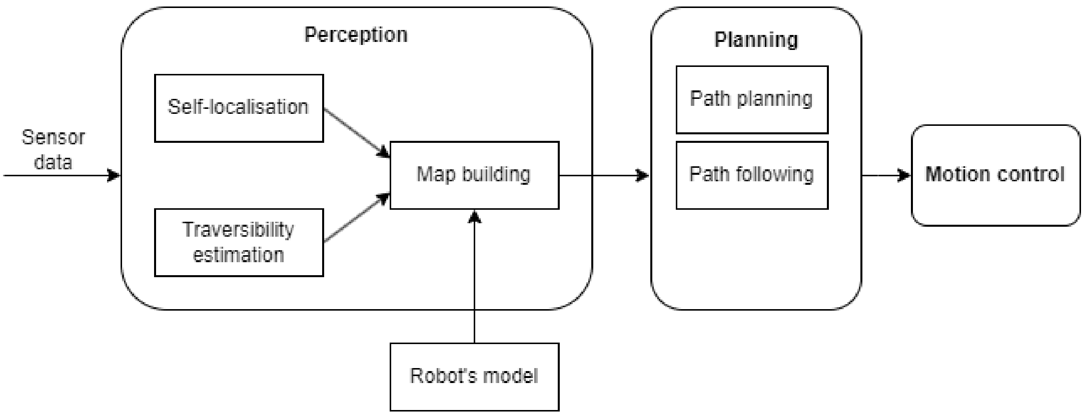


Figure 2.2: Diagram of the tasks that comprise autonomous robot navigation in the context of terrain characterization

2.2 Motion control of wheeled mobile robots

Control in the context of mobile robots refers to the problem of facilitating locomotion in the space they operate in. Control methods are often implemented at the lowest level of hardware, in order to determine the forces and torques necessary to reach a desired destination. In the explored literature, control as a term is referred to in varying degrees of complexities. Whereas some methods only facilitate a low-level control of a mobile robot's motion system, other include trajectory tracking and/or path following which account for high-level control. Finally, all the methods in the following section are considered only in the context of non-holonomic mobile robots.

2.2.1 Classical control methods

Over the years a number of linear and nonlinear control methods have been developed and utilized for various applications in mechanical control systems. Below are brief introductions of the most established classical methods in the literature that utilize analytical models in the context of mobile robots.

Other classical control methods for mobile robots that are not currently presented in this report are Kalman filter and sliding mode control, as those were not employed extensively in the reviewed literature.

PID controller

A common control method in the space of various engineering fields is the PID controller, used for regulation of variables such as flow, temperature or speeds. The implementation of the controller can be composed of up to three distinct control functions, namely proportional, integral, and derivative. A closed control loop is formed where a feedback procedure allows for a continual calculation of the error between a single input and a single output. Consequently a controlled response is applied in order to stabilize the output process (Spong et al., 2020).

PID controllers have been used extensively for various mobile robot applications, and are still widely utilized due to their relative simplicity and reliability. However, their limitation to a single input and a single output and linear nature restrict their use to less complex applications. Shijin and Udayakumar (2017) and Malu and Majumdar

(2014) use PID controllers as a low-level speed control of DC motors for a two-wheel differential drive mobile robot.

Model predictive controller

In contrast to PID controllers, a model predictive controller (MPC) takes into account changes in multiple inputs at the same time. The model predictive control algorithm attempts to control a system by using its dynamics to calculate the optimal future control actions that are within the operational constraints (Findeisen & Allgöwer, 2002). It takes into account the past actions performed by the system within a specified control horizon and predicts changes that can be applied to the variables to reach the desired values. This task is attempted to be done within a specified prediction horizon, and is continually repeated at every time step after a change is applied.

As a more advanced form of a control method, MPC was shown to provide superior motion control performance compared to a traditional PID controller (Rezaee, 2017). Rezaee (2017) also explores the effects of each MPC tuning parameter in applications of non-holonomic mobile robots. However, the work is based on the system's linear model applied to a known and structured environment. Nascimento et al. (2018) state that linear MPC controllers are more mature and successful within a number of applications, but the increased non-linear nature of a mobile robot traversing an unstructured terrain could lead to unsatisfactory results. A number of works explore and implement a non-linear MPC controller for a mobile robot (Hu et al., 2019; Lim et al., 2008; Ostafew et al., 2016). As explored in Park et al. (2015), a traditional numerical approach to optimization of a non-linear MPC algorithm is computationally too intense with regards to off-line implementation. All of the aforementioned MPC solutions therefore employ various machine learning techniques to aid with this issue.

Lyapunov function-based controller

Lyapunov function-based controllers are a class of non-linear control methods based on the use of system specific Lyapunov stability functions. According to Tzafestas (2014b), such a controller can determine if a dynamical system with control inputs can reach an equilibrium and be stabilized. This methodology can be applied with just the kinematics of a system, for example to adjust wheel velocities of a nonholonomic mobile robot (Fareh et al., 2016). However, the same paper also utilizes this method in conjunction with the system's dynamics in order to track and follow a desired path.

Backstepping controller

Dumitrascu et al. (2011) and Zidani et al. (2015) propose controllers for non-holonomic mobile robots based on the *backstepping* control method. In its essence, a non-linear system is broken down into lower order subsystems. First an initial stable subsystem is found that is paired with a Lyapunov stability function to verify system stability. The same procedure is then repeated for each consecutive subsystem that is controlled by a chosen feedback controller. The whole process constitutes a recursive method of going "backwards" to reach an overall stable system. While some non-linear terms can be lost through the process, it provides more flexibility compared to other fully linearized approaches (Vaidyanathan & Azar, 2021).

2.2.2 Machine learning control methods

A disadvantage of the aforementioned control methods is the required knowledge of the model of the controlled system. In contrast, the emergence and wider adaption of machine learning techniques allows for adaptive control where classical methods are improved upon with a relaxed model requirements. Learning based methods are also at the forefront of this area, employing variety of neural networks to more effectively utilize gathered data.

The fusion of classical control methods such as PID or MPC and various machine learning techniques has been explored in order to combine their respective advantages (Carlucho et al., 2017; Hu et al., 2019). Both works present experimental results that perform better than their classical counterparts and ease the process of tuning. In addition, techniques like this can be utilized when it is difficult to obtain a model of robot's dynamics. The capability of obtaining a model of the system's dynamics based on data instead of analytical derivations and numerical calculation is common also among learning based methods that are not combined with traditional controllers (Farias et al., 2020).

Fuzzy Logic-based control

While based on its long history, a fuzzy logic based controlled is often referred to as a part of the classical methods (Maeda et al., 1991). Fuzzy logic controllers operate on the principle of making decisions based on linguistic variables rather than precise numerical values. In the context of mobile robots, these controllers first convert sensory inputs into fuzzy values. For instance, an obstacle's proximity might be translated into categories like 'near' or 'far'. A rule base, crafted from human expertise, then guides the robot's actions. Fuzzy logic is often combined with machine learning (Tzafestas, 2014c), similarly to how other classical control methods are optimized by trained models. For instance, a combination of reinforcement learning and fuzzy logic was used to provide non-linear control for a manufacturing mobile robot (Prabhu & Garg, 1998).

2.3 Traversability assessment

The ability to assess the nature of a robot's surroundings is a crucial part of mobile robots. Their main objectives could be to navigate safely through or around features, while avoiding surfaces that would limit the robot's ability to generate or maintain momentum. The two overarching methodologies in this space are terrain classification and terrain characterization. The process of traversability assessment constitutes the action of map building for path planning purposes.

As mentioned before, the success of a mobile robot in unstructured terrain will depend on its ability to maximize vehicle mobility across multiple surfaces of varying terrain characteristics. This is commonly attempted in parallel with reducing power consumption to carry out locomotion, in order to preserve often limited amount of on-board power storage.

For the purpose of this project, unstructured terrain will refer to an outdoor environment with irregular or previously unseen properties and features. Unstructured terrain can be present in for example off-road areas, planetary surfaces or damaged manufactured environments.

2.3.1 Terrain classification

Terrain classification methods are concerned with identifying and determining the terrain type either during or prior to traversal into a number of candidate terrain classes. The disadvantage of using only classes is limiting the information gained about the terrain at transitions between certain types of terrain classes, where abrupt changes of traversability could occur (Angelova et al., 2007). Additional losses of information could include variations of each terrain class, which can change due to external factors such as weather or lighting conditions. Groups of classes generally consist of surface types (e.g. grass, concrete, gravel, rocks) or of regions described by various thresholds of traversability (e.g. traversable, non-traversable).

2.3.2 Terrain characterization

Terrain characterization is an approach to the same problem of traversability assessment. However, terrain types that can be classified into categories can further vary and behave differently from their nominal state depending on multitude of external conditions (e.g. dry concrete vs. wet concrete). Therefore, it is often not sufficient to only classify a terrain, but it is needed to extract all the relevant features and attributes of a terrain that can affect the mobile robot's ability of motion. The literature encompasses two major methods of accomplishing terrain characterization, namely through the continuous measurement of wheel slip during traversal, and predictions based on visual sensor data made ahead of traversal.

Wheel slip can be defined as the lack of progress of a mobile robot while traversing a surface. The impacts of wheel slip can vary in magnitude depending on the physical interactions between the robot's wheels and the surface (Bekker, 1969a), and cause negative outcomes such the inability to reach a destination or to maintain a consistent motion.

Terrain features are assessed based on key surface characteristics such as roughness, slope, discontinuity and hardness. Slope can be defined as the inclination of the underlying surface with regards to the mobile robot's angular tilt around it's center of mass, and is the main contributor towards tip-over hazards (Seraji & Howard, 2002). Discontinuity refers to steep inclinations of detected surfaces in relation to a reference ground surface. For example, a cliff or a step will have an angle of around ninety degrees from the ground plane a mobile robot is traversing. Other less severe discontinuations of the terrain will be obstacles, often denoted as either positive or negative obstacles. While positive obstacles correspond to features that are protruding from the ground plane, negative obstacles are sunken into the ground plane (Talukder et al., 2002).

2.3.3 Sensors

For the purposes of terrain characterization, there are two main categories of sensors for information acquisition about the surrounding terrain.

Proprioceptive sensors

Proprioceptive sensors capture the states of the robot itself, which are generally used to measure the terrain parameters during traversal. Proprioceptive sensors are used to determine the position, velocity, forces, or energy applied to or by the robot (Martin, 2018). These sensors, which record data like wheel odometry, motor power, and battery voltage, enable mobile robots to directly measure the traversability of terrain they're navigating.

However, these sensors come with inherent limitations. One primary limitation is the need for the robot to traverse the terrain before making any assessments, posing risks of the robot becoming immobilized in challenging terrains. Another challenge is the highly specific nature of proprioceptive data, which captures the immediate interactions of a particular robot with its environment. Types of sensors that fall under this category include motor encoders, potentiometers, gyroscopes and compasses.

Exteroceptive sensors

Exteroceptive sensors are used for remote observation of environments and objects found within them. From the robot's perspective, exteroceptive sensors are mainly absolute measurement sensors, which generally have a lower acquisition frequency than proprioceptive sensors (Papadakis, 2013). Common examples include 2D and 3D lidar, RGBD cameras, stereo cameras, acoustic sensors, and pressure sensors.

The primary advantage of exteroceptive sensors lies in their ability to provide expansive environmental awareness. They enable robots to grasp a broad view of their surroundings, facilitating interactions and navigation in complex terrains. However, the reliance of these sensors on external conditions presents challenges. For instance, a lidar's performance might degrade under foggy circumstances, and cameras could encounter difficulties in low-light conditions. Additionally, the vast amount of data generated, especially by high-resolution sensors such as 3D lidar or RGBD cameras, necessitates considerable computational power, potentially straining the robot's processing capabilities (Borges et al., 2022).

2.3.4 Map building in terrain characterization

Mapping the various factors that could affect vehicle's ability to traverse on a terrain were previously been done in a number of ways. Similarly to control methods, methods of map building that model terrain characteristics can be split into two main categories. Traditional methods employing an analytical approach and numerical calculations, and various methods based around machine learning algorithms.

Analytical methods

As previously mentioned, the topic of terrain characterization for traversability assessment is an especially interesting topic for applications within planetary rovers. Several of the early works propose solutions to better accommodate planetary rovers to the environments present on celestial bodies, and thus mainly focus on prediction and detection of wheel slip.

An early method proposed by Ojeda et al. (2006) to detect wheel slip during a robot's operation is based on the calculation of "motor currents versus rate of turn" (MCR) curves. The hypothesis of the work is that a skid-steering robot on a softer surface will induce wheel slip while performing turning maneuvers, in addition to longitudinal and lateral slip that occurs during traversal caused by wheel and soil interactions. Therefore, the correlation between motor power draw, rates of turn and soil parameters can indicate the amount of slip that's affecting the robot's motion. The data used for terrain assessment is gathered by an IMU unit, and is paired together with motor torque and voltage sensors. The analysis and computation of MCR curves was derived from the founding work on wheel-soil interactions (Bekker, 1969b).

While successful experiments of the paper showed potential applications, this method has the disadvantage of not providing an environment map for navigation purposes. However, since MCR curves contain important soil information of a specific terrain, they can be used to predict potential power requirements or to determine motion resistance (Ojeda et al., 2006). Similar analytical method based around the a similar concept was published by Ishigami et al. (2006), focusing on wheel slip occurring exclusively on slopes. A novel approach of this category for a non-planetary rover was presented in the works of Galati and Reina (2019) and Espinoza et al. (2021).

The challenge of proprioceptive slip detection and estimation methods like those mentioned above, is the inability to avoid areas with high risk of excessive slip (Helmick et al., 2009). Remote knowledge and prediction of potential hazards is therefore a valuable asset that allows a mobile robot to optimize its speed, torque and path, leading to minimized risk and power consumption. This issue was first reported by Angelova et al. (2007), where visual data based on depth imagery is correlated with modeled slip behaviour to generate maps of the local environments. While slip behaviour for any given surface type is modeled by an analytical approach using the robot's kinematics, the imagery data is classified into six classes, provided by a machine learning algorithm.

Broggi et al. (2005) propose one of the earlier exteroceptive methods of traversability assessment for mobile robots, where data from two cameras is used to calculate the *V-disparity* between the left and right camera output, revealing edges of obstacles. Most commonly however, the literature trends to pair the use of imagery data with terrain classification as opposed to terrain characterization, classifying object either by their type or size (Filitchkin & Byl, 2012; Khan et al., 2011).

A more explored exteroceptive method without the use of classifiers utilizes the use of lidar sensors. Liu et al. (2019) perform scans of a robot's surroundings in a polar system, detecting positive, negative or hanging obstacles, as well as straight slopes. The height a slope difference between adjacent lidar points are calculated and indexed by an algorithm into a map of obstacle points, highlighting traversable regions. The differentiation between types of obstacles is done similarly in other lidar based techniques (Larson et al., 2011; Reddy & Pal, 2016). When a planar surface is scanned, the measured lidar points have close to the same values. In the case of positive or hanging obstacles, the lasers of a lidar hit their surface sooner than the surface of the ground plane in the sensor's field of view. For negative obstacles the opposite is true, since the surface of them is further away from the sensor than the observable ground plane. The solution proposed by Lourenco et al. (2020) is partly based on the same standard mapping technique utilizing multiple lidar sensors to build a 2D costmap. However, in this case the cost isn't calculated based on sensory data, either exteroceptive nor proprioceptive. Instead, the mechanical effort that any particular point in terrain could exert on the autonomous vehicle is derived from the vehicle's model. This cost is integrated together with a gradiant that quantifies the slopes withing the original 3D visual data. This information is encoded into a 2D gridmap, which is a common mapping format used by open source robotics software called ROS. This work however mostly focuses on the effects of slopes, and not more variable terrain profiles.

An example of a scenario where visual exteroceptive terrain assessment is not a viable option with regards to obstacles, was explored in the article by Ordonez et al. (2020). The authors propose a combination of a proprioceptive and exteroceptive traversability assessment with a special focus on pliable vegetation. Pliable vegetation refers to obstacles that visually might be considered as non-traversable, but in reality can be interacted with in a safe manner and traversed through. Two different approaches are presented, on two distinct platforms that consist of a lidar and stereo cameras, paired with other equipment respective to their approaches. Upon detection of vegetated areas through its vision systems, the first solution then characterizes an area upon traversing through it. The second solution carries a robotic manipulator to probe an area before a traversal is initiated. The gathered data is used to model the surrounding terrain based on the the dynamics of individual stems of the encountered vegetation. This results in a map in the form of a 3D voxel grid depicting the motion resistance in the robot's vicinity. A separate robot adaptation mechanism is used to train a prediction model based on these grids, in order to quantify the difficulty of traversal through any given vegetated area for path planning purposes (Ordonez et al., 2020).

Machine learning methods

One of the earlier works that introduced the use of machine learning for traversability assessment, published by Howard and Seraji (2001), utilizes a combination of artificial neural networks and fuzzy logic to detect four terrain attributes of roughness, slope, discontinuity and hardness. This is mostly done by edge detection and detection of correlation in image data. This proposal was not tested on an experimental setup, nor does it provide a merged traversability map for path planning purposes.

The exteroceptive sensing techniques for terrain characterization mentioned in the previous section on analytical methods have been further extended by the use of machine learning for map building purposes. While previous solutions generally employed only one type of sensor, Sock et al. (2016) state that this approach may not be sufficient for traversability assessment in unstructured terrain. The authors of that work, as well as Zhou et al. (2022) combine the use of lidar sensor with visual cameras, as both are able to provide complementary information. Zhou et al. (2022) mention that lidar sensors are more proficient at identifying the characteristics of solid structures and cameras provide a way to detect the type of surface that a robot can expect to traverse, and thus also the terrain's attributes. This suggestion is in line with the observation that utilization of only visual cameras is generally applied to terrain classification. However, the two above-mentioned works do not integrate terrain attributes in the generated traversability maps.

A recent approach for traversability assessment paired with path planning employs deep reinforcement learning to generate elevation maps through the use of lidar data (Weerakoon et al., 2022). First terrain features that indicate reduction of stability based on predefined pitch and roll limits are learned in a virtual environment. The learning results are merged with a normalized elevation map computed from raw lidar data into a cost-map. Feasible trajectories are then picked by a least-cost algorithm. Conducted experiments spanned multiple elevation magnitudes across softer and firmer surfaces. Terrain attributes of the various surface types were not taken into account. In contrast, a similarly novel method in the area of unsupervised learning accounts for the effects of the surface being traversed (Sathyamoorthy et al., 2022). The effects of terrain roughness, hardness and surface texture are put into the forefront over the effects of slopes and obstacles. These terrain attributes are extracted from real-world RGB images and correlated with IMU and odometry data in a learning process in order to compute a cost map for path planning purposes. It is stated by the authors that the learning process on a previously unknown surfaces is generally completed in 20 to 25 minutes, lower than comparable methods. The effects of the surface are undoubtedly a major factor in assessment of traversability, as well as the effects of a terrain's slope and obstacles (Seraji & Howard, 2002). An ideal solution would therefore combine each main aspect of the two methods described in this paragraph.

The past and recent works in terrain characterization show that visual and depth information has been a prominent way of gathering information about the terrain. Vulpi et al. (2021) hypothesise that traversability of any given terrain can be assessed exclusively on the back of proprioceptive data in combination with deep neural networks. A number of learning approaches are presented in their work to establish which one of them would be the most suitable for traversability assessment. The learning and experimental processes were conducted across four surface types of a flat profile.

Terrain characterization specific to planetary rovers could benefit from machine learning based methods due to the rich dynamics and high non-linearity of wheel slip in soft soil (Lopez-Arreguin & Montenegro, 2021). This is due to the fact that supervised and unsupervised learning methods were shown to be able to approximate robust non-linear models without any prior knowledge. Alternatively, a novel machine learning approach of modeling soil proprieties (Dallas et al., 2020) proved highly accurate with reduced computational complexity that allows for more efficient implementations in numerical methods such as MPC.

Lopez-Arreguin and Montenegro (2021) express interest in machine learning methods that do not require processing of visual data in order to avoid the higher computational requirements such workloads entail, as it could have important implications with regards to the current constraints of planetary exploration. However, the authors have also mentioned that the commonly employed proprioceptive sensors such as motor currents and IMU cannot adequately represent all types of wheel slip. So far there have not been any proposals for methods that accurately estimate and map traversability parameters in soft soil based on those factors.

Chapter 3

Implementation

The following chapter presents the practical aspects of this project, encompassing design considerations, iterative refinements, applied methodologies, and hands-on procedures. Through this process, this project transitions from abstract principles to the implementations enabling in-field experiments.

3.1 Overview

The aim of this thesis is to explore the challenges, solutions and drawbacks of terrain characterization in unstructured environments for a wheeled mobile robot. One of the main areas of focus will be on discerning the best approach to utilize sensory data for navigation within unstructured environments, enabling the exploration and traversal of unstructured terrains. As stated in the research question outlined in the introductory chapter, two distinct techniques for terrain characterization will be implemented and evaluated in a real-world testing environment. This encompasses both classical analytical technique and machine-learning based technique to address terrain characterization.

The comparison between classical and machine learning techniques for terrain characterization is driven by two main objectives. Firstly, it aims to assess the advantages and disadvantages of each approach. Second, it needs to be researched which one of them is better suited for this task. On the one hand, classical techniques are known to be simpler, more robust, and easier to employ due to the ability to handcraft them for a specific scenario. Conversely, while machine learning techniques may potentially yield superior performance, they necessitate a considerable quantity of high-quality data and an extensive learning period.

The presentation of obtained results, coupled with the discussion, will provide the research community with insights into the prospective trajectory of future developments in this particular area of autonomous navigation.

Once implementation of the necessary hardware and software is in place, field experiments will be conducted at a real-world testing site to assess the efficacy of each software suite. The software covered in this thesis is based on two primary categories: classical methods derived from established literature that has been used in a similar application, and a machine learning approach formulated during the course of this thesis.

This report outlines the methodology for each step that was taken during implementation, testing and comparison. The strategies and considerations taken during testing and data collection are described to give additional context about the progress from start to finish.

3.2 Choice of research methodology

Any potential applications of mobile robots in real-world environments are met with the difficulty of unknown and often changing variables. Such conditions mean that there is a substantial amount of variance from one scenario to another. This is an important consideration when it comes to possible alternatives for the experimental testing required to answer the research question of this thesis. The increased variance in real-world environments naturally leads to an expansive pool of possible elements to test for.

The implemented solutions for terrain characterization capabilities for a mobile robot will be tested in a real-world environment through several field experiments. With the aim of testing in the most unstructured terrain available, an area in a forest will provide a multitude of terrain challenges that a mobile robot would be expected to handle autonomously.

The implemented solutions for terrain characterization capabilities of a mobile robot are set to undergo evaluation in real-world settings via a series of field experiments. The overarching intent behind these tests is to place the system in the most unstructured and challenging terrain. To this end, a forest area has been selected as the testing ground, given its inherent complexities and terrain challenges. Such an environment possesses a myriad of obstacles and conditions that a mobile robot would be anticipated to navigate and manage autonomously in any real-world applications.

The terrain complexities found in real-world environments can be broadly categorized into obstacles such as trees, rocks, and gaps; ground textures that range from flat and rough to slippery; and surface hardness variations, which include soft, hard, and viscous terrains. All the needed terrain characteristics can naturally be found in such environments, which makes it a desired location for testing. This approach is particularly beneficial for the scope of this thesis, ensuring that the findings derived from the testing environment closely mirror real-world scenarios, maintaining minimal deviations. A comparable strategy, involving real-world field experiments with a mobile robot, has been previously employed by Siva et al. (2021).

Real-world testing, while invaluable for its authenticity and depth, might pose a number of limitations. First and foremost, it presents logistical challenges. Securing a suitable environment and managing unpredictable elements like weather can complicate the process. Additionally, real-world tests are often more time-consuming and resource-intensive compared to the alternatives. Unlike in controlled or digital environments, replicating exact conditions for repeated tests can be nearly impossible, leading to potential inconsistencies in results. While real-world data offers unparalleled insight into the actual performance of a system, gathering and analyzing such data can be cumbersome, and there is always a risk of encountering situations that were not anticipated in the design phase. Furthermore, ethical and environmental considerations may also come into play, especially when testing in natural habitats, potentially causing disturbances to local ecosystems.

An alternative method would be to perform tests in a specifically designed testing field. This approach offers higher flexibility, providing more control over environmental variables. However, for the objectives of this thesis, such an approach is not particularly favorable. As mentioned previously, there is significant value in exploring how the technology fairs in a naturally occurring environment to gain a better understanding of the effects of previously unforeseen scenarios. A natural environment, in contrast to the controlled, laboratory-like conditions of a custom testing site, introduces a broader spectrum of variability across all factors influencing a mobile robot's operation.

In contrast to hand-crafted testing sites, testing with the help of physics simulation engines can accommodate the need for testing a large amount of possible scenarios. Given the digital nature of these simulations, a multitude of test runs can be executed either concurrently or over extended durations, surpassing the possibilities of physical tests. This approach not only facilitates the evaluation of numerous scenarios independently but also paves the way for creating a comprehensive digital representation of the environments encountered in the real-world. With the software thatis available, it is feasible to make accurate digital models of the hardware being used during this thesis.

However, difficulties emerge with regards to possible discrepancies when results from simulated physics environments are transferred over to real-world ones. This phenomenon, aptly called the 'reality-gap', can lead to diverse performance outcomes depending on the intricacy of the simulations (Jakobi et al., 1995). With appropriate levels of noise added to the simulation, physical robotic systems have been shown to perform accurately to their simulated counterparts. Further issues arise when the modelled systems interact with other objects through physical contact, which is a complex problem yet to be solved optimally in a simulated environment (Collins et al., 2019). This complication poses great difficulties to solving terrain characterization in simulation, especially since unstructured forest terrain presents numerous obstacles with which a robot must interact.

3.3 Navigation of the robot

For the purpose of assessing each terrain characterization technique based solely on its capacity to discern terrain information, a unified navigation module was implemented for all testing scenarios. This module's design centers around a straightforward navigation mechanism, ensuring no added intricacies are introduced into the overall system. Such a simplified design ensures compatibility with both terrain characterization techniques.

The consistent performance of this module is pivotal to the integrity of the testing process. By operating navigation tasks uniformly across all terrain characterization techniques, it becomes an instrumental tool in guaranteeing a balanced comparison among them. Any potential biases or variations originating from the navigation process itself are therefore minimized, ensuring that any differences observed can be attributed directly to the terrain characterization methods being evaluated.

Each employed terrain characterization technique provides similar kind of output. It is essentially three values of differing scales representing the terrain characteristic as calculated by their own processes. The navigation receives these three values, that can be equated to either turn left, keep driving straight, or turn right. The idea is to attempt to follow that most desirable area.

All of the implemented terrain characterization techniques provide outputs that are analogous in nature, despite being derived from their distinct computational processes. Specifically, each technique yields three values which serve as decision indicators for the robot's movement. They correspond to three possible actions: turn left, continue straight ahead, or turn right. The underlying principle is straightforward: the system uses these values to identify and subsequently follow the most desirable or navigable terrain. Additionally, this decision-making approach serves a dual function as a form of low-pass filtering or smoothing. By prioritizing the most consistent terrain suggestions over sporadic or anomalous readings, the system inherently filters out potential noise or outliers in the data.

Complementing the terrain-based navigation system is a GPS component designed to guide the robot toward its desired endpoint during a traversal. The inclusion of the GPS component aims to refine the robot's trajectory, ensuring that it remains aligned with its destination while navigating through varying terrains. The onboard GPS module provides data that aids in the calculation of the robot's azimuth, an angle in reference due north. Concurrently, the angle between the robot's current coordinate location and the target endpoint coordinate location is computed with the help of a python package. The loop then determines the angle offset by comparing the robot's current heading direction with its azimuth. This offset is ascertained by subtracting the two aforementioned angles. An angle offset of zero implies that the robot is on a direct path toward its endpoint. A positive angle offset indicates a leftward deviation, suggesting the robot should steer left to realign. Conversely, a negative angle offset indicates a rightward deviation, signaling a need for the robot to steer right.

Within the navigation framework, the GPS-derived offset is not just a standalone variable; it serves as a scaling factor that modifies the terrain metrics provided by the terrain characterization techniques. By implementing the GPS offset as a scaling factor the system ensures a consistent influence of GPS measurements across both terrain characterization techniques, irrespective of the ranges of their outputs. The core objective of integrating GPS into the navigation process is to regulate the robot's decision-making based on the end goal. If, for instance, a more desirable terrain would deviate substantially from the end point, the GPS-derived scaling factor will increase the perceived 'cost' or undesirability of that path, thus guiding the robot to consider alternate routes that align better with the target destination.

In essence, the system is programmed to prioritize end-point proximity over terrain quality, but only beyond a certain threshold. Until that threshold is reached, the robot will always favor better terrain, as long as it remains generally oriented towards its destination. The exact cut-off values in terms of degrees of deviation at which the GPS offset outweighs terrain advantage are not known. This dual-priority approach ensures that while the robot aims to reach its end point efficiently, it does not compromise safety by opting for challenging terrains just because they might offer a more direct route.

The implemented navigation system, while efficient for the specific objectives of this research, presents certain limitations worth noting. Primarily, the system is confined to localized decision-making and lacks the capability to consider terrain and navigation challenges on a global scale. This restricts the robot's capacity to make longer-term navigation decisions. Furthermore, even within its local purview, the system does not employ any intricate path-planning algorithms. As a result, it does not anticipate or account for concurrent terrain features, leading to overall sub-optimal navigational choices. The system operates in a more serial manner, reacting to individual terrain elements rather than mapping out a comprehensive path. Lastly, the simplicity of its navigation actions further confines its adaptability by being limited to a predefined turning radius and speed. Such limitations, while acceptable within the scope of this project, highlight areas for potential future improvements and refinements in navigation design.

3.4 Dataset collection

In order to explore the research questions of this project, sensory data will be required for the training of a deep-learning terrain characterization technique. At present, the public domain does not offer any datasets that meet both the volume and requirements specific to this thesis. Given this gap, the decision has been made to generate a unique dataset tailored to our requirements. This data collection will be executed using the current hardware and will be overseen by an operator manually controlling the equipment. This process will also provide an avenue to explore the potential requirements and challenges a dataset collection for terrain characterization application might entail.

There are a number of categories of data recorded by the various sensors:

1. **Stereo Camera Output**:
   - **RGB image stream**: Compressed RGB images intended mainly for user reference that also be be utilized for terrain classification, recorded at 6 frames per second.
   - **Depth image stream**: Processed by the camera, it yields a point-cloud detailing terrain topography, recorded at 6 frames per second.
2. **IMU Data**:
   - **Orientation**: Represents the robot's orientation in space relative to its starting position with yaw being aligned to the magnetic north. The data is in form of quaternion values.
   - **Acceleration**: Captures linear acceleration in the x, y, and z axes, with the influence of gravity filtered out.
3. **GPS Data**: Records the device's geographical location, tying data to specific ground coordinates. The coordinate system used by the GPS module is called *Local tangent plane coordinates*.

4. **Robot's Status**:

   - **BMS**: Monitors battery performance, total energy use and battery voltage during the robot's operation.
   - **Motor status**: Reports rotations per minute as well as the current being drawn by the robot's motors.

To ensure compatibility with the selected navigation system, the dataset's structure is tailored around the prediction requirements of the trained model. Each dataset entry consists of three integral components:

1. **Visual component**: This captures the terrain ahead of the robot, offering an exteroceptive perspective. It provides insights into a yet-to-be-traversed area at the moment of recording.

2. **Action**: At each dataset entry point, the operator determines the robot's direction it takes: left, straight, or right. This direction, set for a predetermined duration and velocity, is called an 'action'. The entire trajectory taken by the robot during this action must be visible in the initial recorded image.

3. **Cost value**: This final component quantifies the challenges encountered during the executed action, providing a measure of the difficulties inherent in traversing that specific terrain section.

In essence, this structured approach ensures that every dataset entry provides a comprehensive snapshot of the robot's interactions with its environment.

The idea behind combining the image data with an action stems from the ability of deep learning networks to discern patterns of movement within an image corresponding to any of the three actions. A trained model would then be used to create predictions for a particular input for each possible action, which is then used for navigation. With this dataset's structure, we also avoid the intricate challenge of aligning what the robot perceives externally via the camera with the data coming from the proprioceptive sensors on-board the robot that records data for that area at a later stage.

One of the main challenges in the progression of this project was in determining the ideal combination of dataset volume, fidelity, and format to produce a reliable model. For visual information, the decision was made to transform the data into a 2D heightmap. To determine the cost value, calculations of energy usage per meter were derived from recorded IMU acceleration data and the motors' reported current readings. While the usage of the cumulative energy spent during an action was an option, this was complicated by the occurrence of wheel slip and mobility resistance caused by obstacles. Such irregularities could distort the dataset, making certain entries less representative when compared to more nominal data entries with minimal external disruptions.

The selected format of data was chosen to be consistent with the classical technique's implementation, ensuring a fair and insightful comparison. While there might be other data formats more optimal for machine learning-based terrain characterization, the lessons learned from constructing this dataset can offer insights applicable to various other structures. Throughout the dataset's development phase, numerous adjustments were made, particularly with regards to image resolution and the calculations of the cost metric.

With the above mentioned structure in mind, the first dataset was created. However, as will be later described, this dataset introduced a number of difficulties during initial stages of model training. The considerations and steps undertaken during each iteration of the dataset's development are elaborated upon in the following subsections.

3.4.1 First iteration of dataset

The primary objective during the construction of the initial dataset was to ensure a comprehensive number of data entries and an extensive representation of the test environment. Entries originating from most locations within the test area were included, regardless of their complexities or viability for the use with a wheeled mobile robot. This included not just optimal paths but also unfavorable scenarios, such as instances where the robot might become trapped due to an obstacle or attempting to ascend a slope beyond its capability. This strategy was used in an effort to cover a wide array of potential real-world challenges. While efforts were made to ensure diversity by revisiting similar terrains multiple times, the emphasis was on maintaining randomness. This extensive data gathering resulted in a dataset comprising roughly five thousand entries.

The first dataset encountered several setbacks, with the most notable being inadequate training performance due to a high tendency towards overfitting. The complexity stemming from the high-dimensional raw sensor data was a major contributor to these problems. Additionally, the use of IMU acceleration data as the basis for distance calculations might have incorporated inaccuracies, hindering the network's capability to generalize effectively. Even with multiple training attempts using this dataset, the achieved accuracy was insufficient for the demands of autonomous navigation.

Another key consideration with the initial dataset was the sheer volume of data entries required to capture the vast dimensionality of the environment. Preliminary experiments using synthetic data (detailed further in chapter ??) hinted that the dataset size might not need to be overly extensive. Although both datasets — synthetic and real-world — aimed to leverage edge detection as the primary tool for identifying major terrain changes, the real-world dataset posed a unique challenge. While higher pixel values did signify elevated terrain, the cost attributed to each pixel was not as straightforward to determine as in the synthetic dataset. This complexity was exacerbated when considering the noisy and inconsistent readings from proprioceptive sensors, such as motor currents and IMU-based distance estimates. Such complexities hint at the potential need for a significantly larger volume of real-world data entries, compensating for the inconsistencies and detail loss.

In summary, the volume of training data required for a machine learning system to consistently perform in real-world scenarios surpassed our initial estimates.

3.4.2 Second iteration of dataset

In an attempt to implement a working solution to answer the research question presented by this thesis, a second, revised dataset was created.

The guiding principle for the second dataset shifted from emphasizing data volume and coverage to emphasizing the consistency and applicability of each data entry. In other words, a greater care was put towards recording data entries that were only much more beneficial towards a viable navigation strategy. This entailed deliberately excluding entries that portrayed scenarios where an autonomous system would face considerable mobility challenges. Simultaneously, there was an emphasis on ensuring consistent nominal mobility, meaning instances where there was significant wheel slip or resistance at the beginning of data recordings were minimized. A combination of these adjustments not only delineates the dimensionality to capture the most relevant information, but also mitigates factors that could introduce inaccuracies in the dataset's numerical attributes.

In an effort to further address issues with excessive dimensionality, it was decided to reduce the resolution of the final images from 32x32 to 16x16. While validation tests highlighted that this solution, though beneficial for training performance, had its own set of drawbacks. Specifically, the diminished resolution hindered the model's ability to discern subtle variations in the terrain. While it performed well in areas with larger, more distinguishable obstacles present in the input images, it failed to recognize and adapt to smaller terrain variations. This discrepancy was particularly pronounced on flat gravel terrain.

As learned from initial training done on this dataset, the reduction of image resolution can indeed simplify the training process, and it can simultaneously compromise performance, especially in specific terrain profiles. A deeper dive into determining the precise resolution and dataset size that balance detail accuracy and computational efficiency is crucial.

3.4.3 Processing of 3D point-cloud data for deep learning

The robotic system is equipped with a single stereo depth camera, which generates a continuous stream of 3D point-cloud data. Each point within this cloud has coordinates relative to the camera frame, along with a depth measurement, representing the point's distance from the camera lens. Both of the terrain characterization techniques employed in this study leverage this information. However, each method processes the data differently and requires in distinct formats for their respective inputs.

The classical terrain characterization method directly employs the 3D point-cloud without requiring additional pre-processing. In contrast, when using a deep learning network, as intended in this study, converting the 3D point-cloud into a 2D image proves beneficial. While the camera does produce 2D depth images, at the time of implementation it was desirable to have image input that was consistent with the input used by the classical technique and was compatible with implemented navigation. This is namely a top down representation of the area in front of the robot with that displayed the height of each position relative to the ground.

With this in mind, a concept of a 2D heightmap image where each pixel is encoded in the grayscale range of 0-255, was implemented. The starting pointcloud of each recorded data container from data collection phase is taken, and processed by a C++ program. This program utilizes the PCL library with various point-cloud processing functions. Once the correct point-cloud is acquired, it undergoes a series of transformations. An integral step is voxelization, where the continuous 3D space is discretized into voxel units. This process, akin to the pixelation of images, serves dual purposes: it significantly reduces the data's density, thus improving processing speed, and filters out potential noise, ensuring that our visualizations are both efficient and accurate.

Further refinements of the data is done through a cropping function. This is exclusively done to remove the parts of the point-cloud that do not provide any relevant information, and only keep the primary region that can effect the system. At the same time, it also reduces the dimensionality that would otherwise be put into a deep learning network.

With the point-cloud now in an optimized state, the next step is translating its 3D coordinates into a 2D plane. This is achieved by calculating the pixel's x and y positions of each point relative to the spatial dimensions of the point cloud, and then scale that position according to the resolution of the desired heightmap. This is calculated with the following formulas:

$$x_{pixel} = \left(\frac{x - x_{min}}{width_{pc}} \right) \times (width_{img} - 1) \tag{3.1}$$

$$y_{pixel} = \left(\frac{y - y_{min}}{height_{pc}} \right) \times (height_{img} - 1) \tag{3.2}$$

This proportional transformation inherently possesses the potential to merge multiple 3D points into a singular 2D pixel, essentially averaging their properties. This ability of downscaling the original data to any resolution is especially pivotal in reduction of dimensionality, and in addressing empty voids in the original data, as the merged values of neighboring points provide a plausible value for these missing regions.

The last step of the conversion is the calculation of each pixel's value. In the initial approach, pixel values were directly derived from the depth data present in the raw point-cloud. This simplistic method, however, led to images with a pronounced gradient. The bottom pixels, being closest to the robot, had values starting near zero, while the pixels at the top of the image, representing points furthest from the robot, had values nearing 255. Such a gradient-rich representation complicates the differentiation between varied height values, not providing enough contrast in regions of interest. This was perceived as a potential issue when considering these images as a dataset for deep-learning, where distinct features could be critical for model accuracy and generalization.

Instead, a height value is calculated similarly to how it is done by the classical technique, utilizing plane-fitting and a perpendicular point-to-plane distance calculation. This also contributes additional consistency between the two techniques that are being compared. Upon calculating the height, each value is mapped to a gray-scale gradient that is proportional to a predefined range. With 0 (black color) representing the lowest limit and 255 (white color) the highest limit, every elevation is now visually encoded at each pixel. The resulting 2D heightmap can be viewed in figure 3.1.

(a) (b)

Figure 3.1: The same location viewed as: (a) RGB image (b) 2D heightmap

In conclusion, this program enables an orthographic projection of the original 3D point-cloud coupled with value encoding that is derived from the original values.

3.5 Classical terrain characterization technique

The classical terrain characterization techniques chosen for this thesis is called Terrain characterizer (Nygaard et al., 2021). While it is originally developed for a mammal-based quadruped robot, it can be adapted to be used in applications with a wheeled mobile robot. The primary intended function of this framework is to aid a morphologically adaptive quadrupet robot with the decision making when traversing various terrain types such as concrete, sand and gravel.

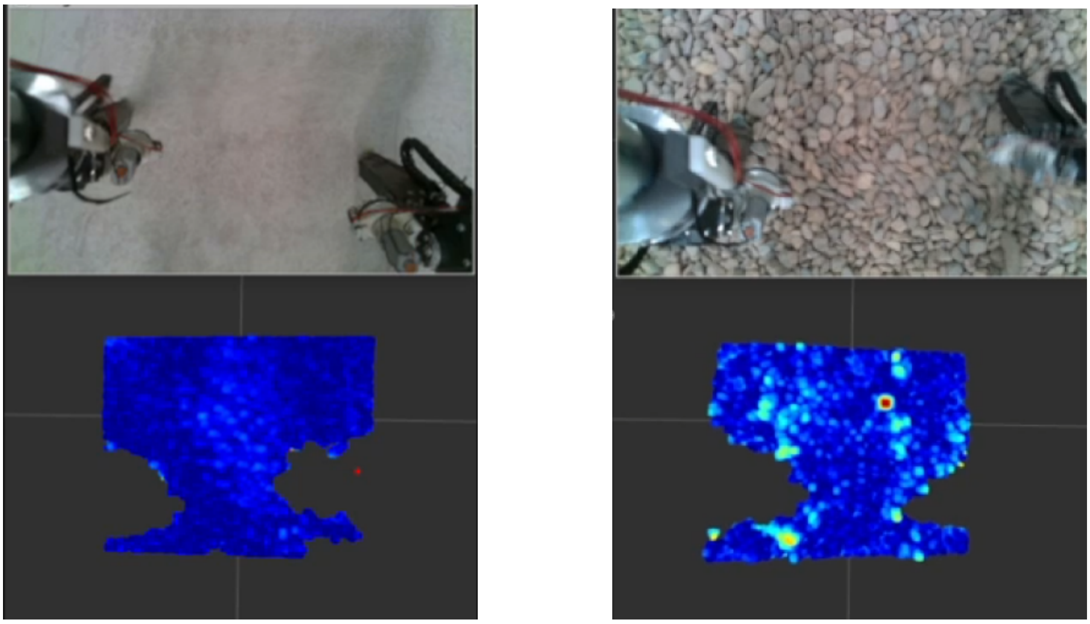


Figure 3.2: Example of a roughness estimation by the DyRET terrain characterizer

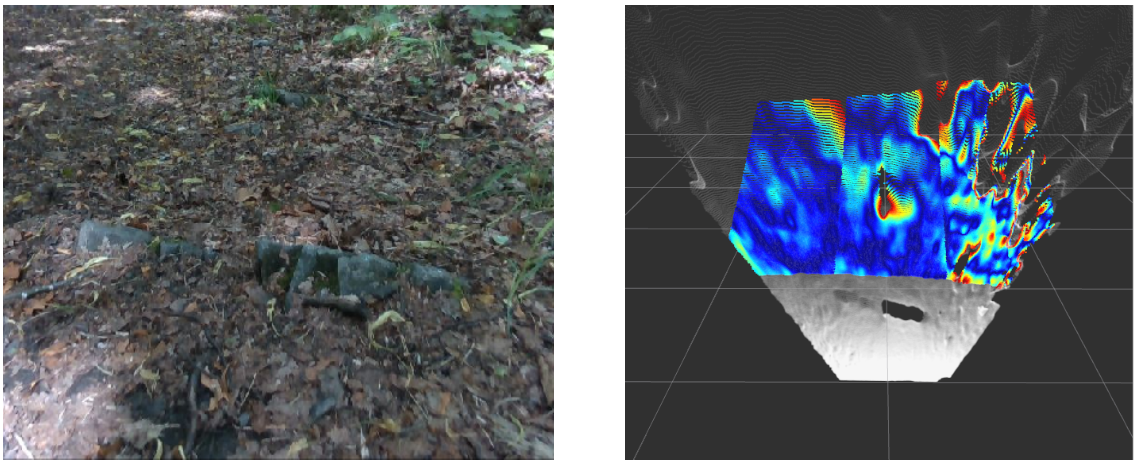
The primary objective of this algorithm is to estimate the roughness of the terrain. Input for this algorithm is a 3D representation of the ground ahead of the robot, generated by a stereo depth camera in the form of a 3D point-cloud. Each point within this point-cloud denotes its depth, which can alternatively be interpreted as the distance from the camera lens. Subsequently, a plane-fitting algorithm is deployed on this point-cloud, delivering a best fitting approximation for a ground plane based that fits the collected data. The algorithm employs a technique known as Random Sample Consensus (RANSAC). The resulting plane is determined by its coefficients that serve as a reference against which the height of each point in the cloud is measured. The subsequent measure of height is given by the equation:

$$d = \frac{|a \cdot x + b \cdot y + c \cdot z + d|}{\sqrt{a^2 + b^2 + c^2}} \tag{3.3}$$

Where:

- (x, y, z) are the coordinates of the point.
- a, b, c represent the coefficients of the plane's normal.
- d is the plane's offset from the origin.

The magnitude of this distance provides a direct measurement of how far the point lies from the plane, regardless of whether it is above or below it. In its application for the classical terrain characterization technique, the derived height value is subsequently normalized into a color space that can be used as a visualization (fig. 3.3). While this approach might pose limitations when applied to non-flat terrains, it offers a means to estimate the elevation of each point relative to the fitted plane, whether above or below.



(a) (b)

Figure 3.3: The same location viewed as: (a) RGB image (b) Terrain characterizer point-cloud

The processed point-cloud now represents points with height values instead of depth. The concluding value for a specific area is derived as the mean of the squared distances from the plane-fitted ground. This serves as a roughness estimation, with greater values signifying increased irregularities in the ground's surface. An example of a representation of roughness can be seen in figure 3.2 above.

Additionally, a maximum distance parameter from the flat surface is applied across the data points. This distance functions as a filter to cut out every area from the observed environment that is above the predefined maximum. Originally, the main application of the filter was to cut out the robot's legs from the observations. However, this approach also doubles as a way of close proximity obstacle detection. By finding the most optimal value for the maximum distance parameter, any undesirable gaps and steps can be filtered out, and essentially perceived as areas to be avoided. In the final configuration, this parameter was set as the radius of the robot's wheels used in this implementation. The reasoning for this value was based on the idea that if an obstacle's absolute height exceeds the wheel's radius, the robot faces significant wheel slip and often risks getting stuck.

For the purposes of this project, modifications were introduced to the original process to better align with the specific use-case and the integrated navigation system. While the core mechanism of calculating fitted planes and determining the height of each point within a designated area remains intact, the application loop has been tweaked.

The incoming pointcloud is divided into three equal vertical segments within the predetermined field of view. This segmentation allows each section to undergo its own plane fitting to reduce the amount of inaccuracy on a non-flat terrain. The resulting pointcloud that contains the calculated values of each point is also cropped to better represent the available operational area of the robot. The three sections for left, middle and right of the resulting pointcloud and their values are used by the navigation to decide which direction is the most optimal.

The original work also includes hardness detection (Nygaard et al., 2021). This capability is realized by a couple of proprioceptive force sensors located in the legs of the quadruped robot. Hardness detection on a wheeled robot would require a different implementation, and therefore the hardness estimation of this software will not be used during this thesis. This framework therefore only relies on visual-based exteroceptive sensing.

3.6 Deep-learning terrain characterization technique

To ensure a fair comparison between the two techniques employed in this project, specific functionalities concerning the machine learning method were required. While the input used by this method could be different based on its processing requirements, it was necessary for the output to be compatible with the navigation system in use. Therefore, the model was designed to: receive real-time input data of the terrain ahead of the robot; process this data offline, and subsequently output the cost for each of the three section used for navigation. The most important function of the model is that the section of an image with the lowest cost signifies the most favorable terrain.

With this understanding, each data entry fed into the network comprises two components: an image in the form of a 2D heightmap and an associated action taken at the moment the image was captured (either left, middle, or right). The anticipated output from this pairing is a value denoting the energy consumption per meter traveled during the duration of the action taken. By integrating both an image and a non-spatial variable, the network predicts the energy per distance used for each potential action.

3.6.1 Proof of concept using synthetic data

Prior to delving into the intricacies of real-world data, a proof-of-concept phase was undertaken, utilizing a synthetic dataset. One of the intentions behind this phase implementation was to familiarize with the nuances of the technology, therefore enhancing the level of preparedness when transitioning to real-world dataset. The synthetic data was attempted to replicate the expected visuals of the real-world dataset in order to facilitate the creation of a preliminary machine learning network.

While the resulting models were not overly specialized to train on this synthetic data, it provided essential validation insight. Instead, this phase served as a testing ground, assessing how effectively the chosen data structure could be used in a deep learning network. Moreover, the experimentation shed light on potential requirements for a successful terrain characterization model, particularly emphasizing dataset size, image resolution, and other training specific parameters.

Synthetic dataset

Given that the real-world dataset would use 2D heightmap images encoded in greyscale, we generated comparable imagery using various randomization and noise-introduction techniques available in both OpenCV and numpy packages. Techniques such as noise addition, Gaussian blur, displacements, and distortions were employed to simulate pixel values that mirrored those anticipated in real-world data. For instance, pixel values nearing 255 signified positive elevations from the ground level, values approaching 0 indicated the opposite, and the majority of the image hovered around middle of the spectrum.

The preliminary network proved adept at modeling the synthetic dataset effectively, and exhibited the behaviour needed for the employed navigation. Notably, this result was observed starting with a mere hundred images, with incremental enhancements seen up to a dataset size of one thousand images. However, this was accomplished using a simplified replacement for the cost value of each pixel, yielding a highly accurate terrain representation. Such accuracy would most likely not directly carry over when applied to sensor data from the real world.

3.6.2 Initial model design: A conventional CNN approach

The first design of the deep-learning network leveraged a straightforward Convolutional Neural Network (CNN) architecture, built with the assistance of PyTorch, a popular deep learning framework. Initially, the model demonstrated proficiency in modeling synthetic data, handling the similar but much more accurate dataset with ease. However, challenges arose when the same architecture was subjected to a real-world dataset. Despite rigorous efforts, the model consistently failed to extrapolate meaningful learning outcome from this dataset.

The final version of the architecture of the initial model was based on four convolutional layers. Each of these layers was constructed sequentially to facilitate a convolution operation, followed by batch normalization layer. Subsequently, the Rectified Linear Unit (ReLU) activation function was introduced to introduce non-linearity and address potential vanishing gradient issues. Lastly, a max-pooling layer was applied, aiming to downsample the input representation and reduce its dimensionality, thus making computations more efficient.

After these sequential convolutional stages, the forward propagation takes the heightmap data and reshapes it to align with the architecture of the fully connected layers, transitioning from a 2D spatial representation to a 1D vector format. Parallel to this, the action input, representative of the driven direction associated with the heightmap, is processed through its designated linear layer. The model then combines these two distinct representations by concatenating them. This integrated data is then passed through additional dense layers, driving the model towards its final objective: predicting the energy consumption per traveled meter associated with a specific heightmap-action combination.

Despite experimenting with varying numbers of these layers, the overall network performance remained largely unchanged. Several strategies were explored to optimize this architecture further. One such approach was the introduction of data augmentation techniques. However, the application of augmentation failed to bring any substantial improvement to the model's performance. Parameter tuning was predominantly guided by an iterative process of experimentation. Although some configurations delivered marginally better outcomes than others, the persistent issue was the model's inability to train satisfactorily on real-world data. The hypothesized factors that could cause the poor performance of the architecture were high dimensionality and data quality.

The primary dataset exhibited a high dimensionality relative to the quantity of available images. This discrepancy can lead to challenges in training, as a high-dimensional input space requires an extensive and diverse dataset for effective learning, and the lack of ability to generalize throughout the dataset. Furthermore, another challenge was the quality of the data, specifically the presence of possibly inaccurate values for the predicted variable. Such inconsistencies could be making it harder for the model to discern patterns.

Given these challenges, a decision was made to overhaul the dataset. New guidelines were set in place to curate a dataset that was dimensionally constrained, while eliminating as many factors affecting the accuracy of the predicted variable. These guidelines and the subsequent process are elaborated upon in section ??. While the revised dataset resulted in enhanced model performance, the challenge of overfitting persisted, albeit appearing in later stages of the training epochs.

3.6.3 Exploring a ResNet based model: An experimental pivot

In the realm of deep learning, it is often stated that when one approach falters, there are several others to be explored. Given the challenges faced with the traditional CNN model, an experimental pivot was made towards a more modern, and often cited, architecture in deep learning literature: the Residual Network or ResNet.

ResNet's core innovation lies in its "residual blocks". These blocks allow the network to learn residual functions with reference to the layer inputs, rather than re-learning the entire transformations. This is achieved via "skip connections" or "shortcuts" which bypass one or more layers. This mechanism assists in addressing the vanishing gradient problem, a challenge often encountered in deep networks where gradients tend to diminish in magnitude as they propagate through layers, leading to ineffective training.

In the context of this specific project, a simplified version of ResNet was adopted. The primary objective behind this transition was to get a network that would be able to process the dataset at hand and provide a model that would enable navigation of the robot. As this type of architecture is specialized for applications where deep learning models normally have issues with overfitting, it seemed like a suitable option to experiment with.

In our tailored ResNet network, the main idea is the incorporation of residual blocks. Each of these blocks, encapsulated in the ResBlock class, comprises of layers similar to those in the initial CNN based network design. Instead of solely relying on the outputs of these layers, ResNet introduces a "shortcut" mechanism. This means that the input to the block is added (or "shortcut") to the output of the block, resulting in what is called as a residual connection.

Preliminary attempts with the ResNet-based model showcased a significant improvement in training capabilities. The model acquired from this network was the final model used for the final experiments.

3.7 In-field consistency and management

Earlier, it was highlighted that field experiments would be conducted in a forest area in Viken, Norway. This environment presents an unstructured terrain with a wide selection of terrain features. However, in the initial stages, to ensure effective prototyping and iterative testing, simpler and more controlled testing grounds were employed. This approach ensures that specific functionalities of each system are validated before advancing to more complex scenarios.

Once each function of the mobile robot is validated, all subsequent testing geared towards collection of results was completed in one general testing area, while ensuring each separate experiment was always carried out in its unique location. Areas that were previously used for dataset collection were excluded to ensure unbiased results. The experiment locations incorporated an array of naturally occurring obstacles of diverse scales, from trees and rocks to branches and fallen vegetation. It is critical that these obstacles offer both traversable and non-traversable challenges for the mobile robot, simulating a realistic environment. In addition to obstacles, the testing fields should consist of different ground types. Each ground type has their own set of characteristics that determine its texture, hardness or roughness - such as grass, gravel or fallen vegetation. The topography within the designated test boundaries should present varied slope gradients, introducing different levels of risk for the robot. Examples of valid locations for experiments are depicted in figure 3.4.

Figure 3.4: Examples of valid locations for experiments

Finally, it is crucial to acknowledge the role of external conditions. Weather, for instance, can significantly impact the technological components during experiments. To maintain consistency and avoid unwanted influence from weather-specific factors, all experiments were carried out outside the winter season and during periods without rainfall.

To ensure consistent field experiments across different software variations, it is crucial to manage the potential environmental changes caused by repeated experiments located in the same location. Each subsequent experiment run, especially if initiated from a common starting point, might alter the environment slightly, and possibly introduce unwanted variation in the data. Tests were therefore carried out by alternating between techniques with each successful run, in order to spread out the effects of deterioration introduced by the robot.

The planned test procedures for each terrain characterization technique was to perform several traversals from a chosen point A to a chosen point B. Each intended experiment on a specific configuration was carried out multiple times to verify if the results were statistically significant, with at least three tests being the minimum requirement to be part of the results of this project. Results that varied greatly between each test run on the same test configuration were not considered to derive conclusions from.

Effectively addressing the research question based around its limitations proposed in this thesis relies on making the most fair and representative comparison between classical and machine learning terrain characterization techniques. Since there are considerable differences in how both operate on the fundamental level, each type of technique should be affected by the least amount of external variables possible. A considerable difference between classical and machine learning ways of computing any problem, is that the former is developed based on human understanding of the problem. However, deep-learning, while developed and often also inspired by processes that are common to humans too, fundamentally solves problems in its own understanding. While the underlying principles of deep-learning based solutions are often unexpected and abstract to humans, the final outcomes provide an insight that can inspire further implementations.

To ensure a balanced comparison between the classical and machine learning techniques, biases inherent to each approach should not influence the other. Therefore, the testing of machine learning techniques should be conducted only after completing classical technique testing, to make sure the latter is only based on the operator's initial bias. Similarly, data intended for use in machine learning was collected prior to any experiments, in order to not introduce new biases into manual data collection from seeing autonomous solutions. This order of procedures for data collection and experimentation also contributes to form the most generic machine learning solution possible. However, it is important to note that data collected by manual control will not be completely free of bias, as there is certain amount of bias inherently with any human operator. Essentially, the main consideration with regards to data collection is to implement classical solutions and machine learning solutions that are not inspired by each other.

3.8 Ethical issues

Since this thesis also describes the analysis of the datasets accumulated throughout the outlined methodology, as well as present the final results, it is important to take a look at any ethical considerations.

Stewart (2011) presents discussion on a number of potential ethical issues within the process of scientific research. First and foremost, the validity of research that is based on first-party data and analysis is dependant on their quality and honesty. In the context of this thesis, the operator that is manually acquiring the data necessary for machine learning purposes, must be aware of the biases they can introduce into the data. This is an important factor that needs to be minimized wherever possible, such that the research questions of this thesis can be answered based on the most representative foundation.

As the person analyzing the datasets and conducting the research will also form a certain hypothesis before the field experiments are conducted, it is paramount to take into account that the actual results could differ from the hypothesis. Therefore, the author should not focus on steering the experiment in such a way that the datasets would give the same results as initially hypothesized. Being aware of this is important, since unexpected results with appropriate analysis and discussion are still valuable to the research community.

In a similar fashion, during the search for the real-world testing fields to conduct experiments on, it is important to not pick areas that would compose the absolute best case scenarios. This could potentially not represent the real-world performance, as it would be closer to a controlled experiment. To reiterate, one of the main goals of this thesis is to challenge the terrain characterization technology on the same scale as in any arbitrary robotic mission. Therefore, the chosen terrain should be chosen to represent a wide variety of features that can be reasonably expected to test and assess the main factors of terrain characterization techniques.

Chapter 4

Experiments

The subsequent chapter delves into a detailed presentation of the experimental setup, offering a comprehensive overview of both the hardware and software components utilized. Below is also an overview over the experiments that will be carried out. For each of these experiments, we'll outline their primary objectives, anticipated outcomes, and the specific steps or actions that constitute them.

4.1 Experimental setup

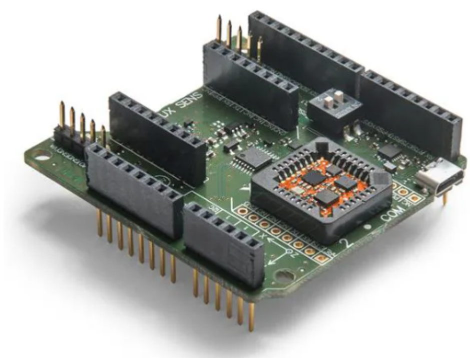
The platform used for the experiments is a wheeled mobile robot, specifically the AgileX Scout Mini rover (fig. 4.1). This model is categorized as a high-speed, all-terrain four wheel non-holonomic mobile robot. It employs a differential drive and an independent suspension, utilizing all four wheels to perform steering and rotations in place.

Figure 4.1: The robotic platform with installed hardware

The main consideration in the choice of this platform is its potential capability in an off-road terrain. Optimization of the performance tied to the hardware provides the most fair and consistent comparison only between the software used during each test run.

The sensor suite mounted on the robot will consist of a stereo depth camera Intel RealSense D435, an IMU and a GPS unit (fig. 4.2). A combination of these sensors is used to provide capabilities of perception in the local region in front of the robot, capability for navigation towards an end goal, and a way of measuring the forces applied on the entire system.

(a)



(b)

Figure 4.2: (a) = Intel RealSense D435, (b) = XSens IMU+GPS module

The stereo depth camera has a relatively narrow field of view, but for the application needed in this project, it will provide a detailed high resolution data to sense finer variations of terrain in the robot's close proximity and trajectory.

For any capabilities at a longer range, a LIDAR sensor would need to be considered. It often has the wide field of view and distance needed to provide meaningful information at a more global scale, such as detecting any obstacles that should be accounted for when planning a global path. The relative difference between capabilities of the two types of sensors is illustrated in figure 4.3.

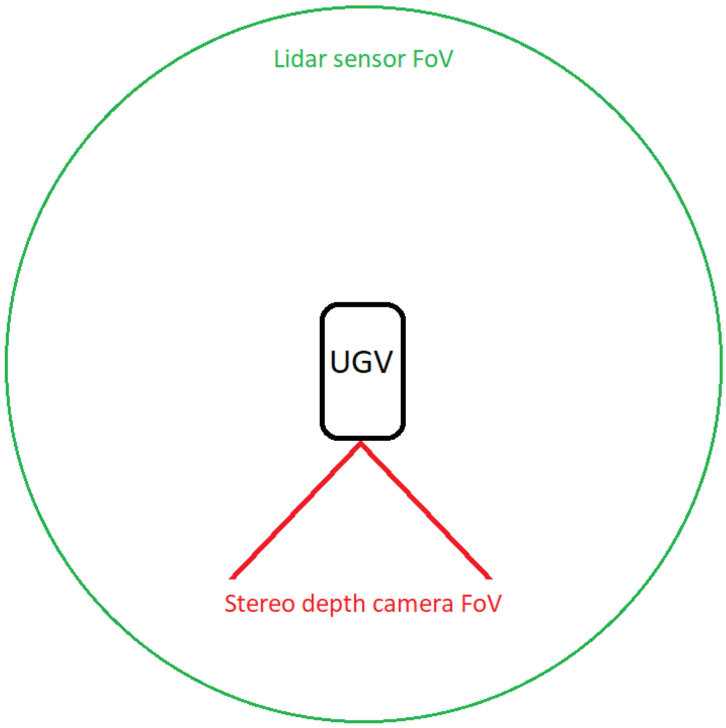


Figure 4.3: Fields of view of the two visual sensors (not to scale)

An IMU module delivers orientation and acceleration data. The orientation aids navigation, while acceleration quantifies the terrain's impact during traversal. The robot also features sensors that monitor internal operational states, including wheel odometry, motor speed, current, and voltage. A GPS unit is also present which reports the physical location of the system expressed in coordinates. The specific coordinate system used by this particular GPS is 'Local tangent plane coordinates'.

The entire system is powered by the on-board battery with nominal voltage starting at 29 volts. As this voltage drops off with reduced charge, a DC to DC regulator is installed to provide a steady 12 volt output for installed computer.

4.1.1 Software

All the necessary terrain characterization software for this project will be integrated using ROS Noetic. The computer that is present on the robot that is responsible for data processing, and executing the autonomous systems, is an Intel NUC running on Ubuntu 20.04 operating system.

The developed of the mobile robot provides a ROS package, which facilitates communication via the CAN to USB interface and offers ROS topics and messages for interfacing with the hardware's data outputs. While the core package remained the same, some customizations were made, such as the inclusion of custom messages tailored for navigation. The developer's original package code is accessible on their GitHub page. Additionally, the platform's sensor hardware utilizes their respective drivers and ROS packages provided by their manufacturers.

In the course of this project, several additional Python and C++ packages were leveraged to cater to different elements of our implementation. Below is a rundown of the most notable packages:

- PCL
- OpenCV2
- PyTorch
- Numpy
- Geographiclib

The final source code that was used to carry out experiments, train a model and to process all required data can be accessed on a public github repository [1]. This includes each C++ program and python scripts, as well as modified version of the original code that was adapted to implement the classical terrain characterization technique.

[1] https://github.com/erikskul/acit5930_23

4.2 In-field experiments

These following experiments were conducted with the main function of exploring the system's capabilities and quantifiable performance in various scenarios. A secondary goal was to generate data that offers a comprehensive view across all test conditions. Two primary criteria will dictate location selection for the experiments. The first main category of experiments will be aimed to show the capabilities of each system in specific scenarios. The other category of tests would be intended to illustrate the overall performance over a longer distance, which will be more comprehensive in nature.

Experiment A

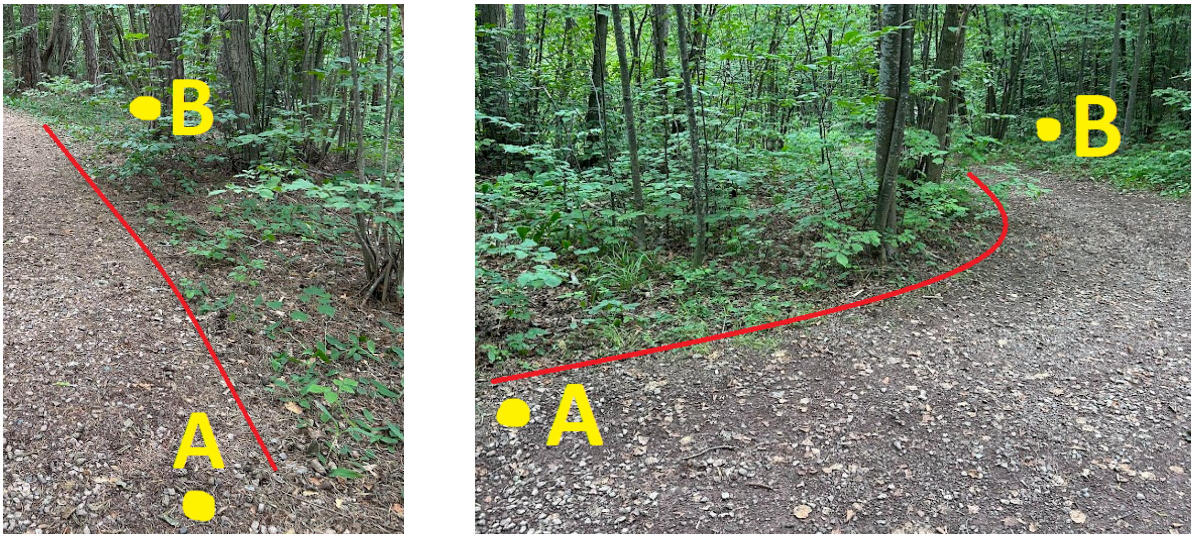
First set of experiments is intended to be a brief controlled experiment, done over two different types of terrain, namely gravel and grass. This experiment is not executed by either of the autonomous systems, but instead is only done by a human operator in a straight line with no additional input. To keep as many variables consistent, each instance of this experiment was conducted on the same incline, exclusively downhill to minimize wheel slip, and across an identical distance. The primary goal for this experiment is to validate if the differing terrain types lead to varied energy consumptions. The expected outcome from the data gathered from these experiments should show that traversing the same distance on the gravel terrain is more energy efficient than compared to traversing on grass.

Experiment B

The second category of experiments focuses on the robot's capability to navigate through a terrain transition, specifically from gravel to an unstructured forest terrain. The robot starts from a point A placed on a flat gravel trail, and is required to correctly identify the most suitable entry point through the terrain transition situated perpendicular to the transition's line. Upon entering, the system must navigate through the terrain's variations to approach point B. Finally, the point B is placed past a similar terrain transition as at the start of the experiment. The primary aim of this experiment is to evaluate if the if the navigation system is able to steer the robot towards the end position while still making reasonable decisions regarding the local terrain.

Experiment C

The third set of experiments was aimed to confirm if the system is able to follow along a contrasting terrain transition (fig. 4.4). The point A is placed on a flat gravel trail, while point B is placed in an unstructured terrain, leading to an optimal path that is more or less parallel to the transition line. Ideally, the system should prioritize the gravel trail — the more favorable terrain — only crossing into the unstructured terrain as necessary to reach point B. A couple of variations of this scenario will be undertaken, with various starting orientations and transition line curvatures to provide varying difficulty levels. This includes a starting orientation that is parallel to the transition line, a starting orientation that is perpendicular to the transition line, a transition line that is straight, and a transition line that curves. The last mentioned variation provides a specific scenario where the system needs to turn towards the point B, but simultaneously, the more desirable gravel trail expands to the opposite direction.



(a) (b)

Figure 4.4: Examples of *Experiment C*. Red line = transition line between two terrain types

Experiment D

Fourth category of experiments is a typical scenario that can be often encountered in a forest terrain. It is executed in an area with thick vegetation and unstructured terrain, with a clear trail going through that environment. Initiated at the trail's start with point A, and concluding at a predetermined distance on the same trail with point B. The system is expected to follow the clearly visible trail, avoiding the challenging terrain on either side of it.

Experiment E

Fifth category of experiments is intended to test each system in areas with higher amount of obstacles of varying sizes. Both point A and point B are placed inside of areas with only unstructured forest terrain. The main objective behind these experiments is to see how each system identifies and reacts to varying sizes of obstacles, and if they can successfully navigate through a larger stretch of solely unstructured terrain. This scenario also provides the best opportunity to reveal the differences in the recorded roughness by each system, quantifying their ability to adapt to terrain variations.

Experiment F

The final category of experiments are long duration tests that aim to give the closest approximation of a real-world use case for a robotic system. Compared to the other experiments, the distance between point A and point B is intended to be considerably extended. The main goal behind these experiments is to gather data that holds more significance due to the increased duration, but also to see if and how the two autonomous systems are able to reach the point B, without focusing on a specific terrain scenario.

Chapter 5

Results and discussion

In this chapter, the gathered empirical findings from in-field experiments will attempt to answer the research question of how a deep learning-based terrain characterization approach compares to a classical analytical one when employed in the real-world. The data consisting of energy consumption and IMU based metrics will be presented, followed by relevant discussions and observations. Each navigational mode- whether human-operated, classical technique (TC system) or deep-learning technique (ML system), will undergo an individual assessment, analyzing their capabilities and performance. Where relevant, comparisons will be drawn between them. Through a combination of quantitative metrics and qualitative insights, this chapter seeks to discover the capabilities of the systems in real-world scenarios.

5.1 Structure of results

The collected data offers a quantitative overview over energy consumption and the ability to asses terrain associated with each mode of navigation used throughout all experiments. While the primary focus of our analysis is the comparison between the two autonomous systems, we also consider results from tests operated by humans to provide a context as it's a common way of operating robotic systems.

Each technique will be evaluated by a number of criteria that are related to the mobile robot's ability to traverse through an environment. Each criteria represents a different capability that might be desirable in any real-world use case, namely: total time used to finish a traversal, total energy consumption of a traversal, total energy consumption per distance travelled, and a measure of roughness that would indicate the difficulty of the terrain taken throughout a traversal. The measure of roughness in this context is the summed average of IMU data's standard deviation, specifically the pitch and roll components (lateral movements in the x and y directions). This means that the roughness metric would increase when the robot takes a path through a more varied terrain, as more vibrations and inclines are effecting the system throughout a run.

To facilitate the analysis of gathered results, we've categorized the data into two primary structures based on the terrains most frequently encountered throughout the testing phase. Every valid experiment conducted, irrespective of its duration or distance, finds representation in these categories. The first category, labeled as "Mixed terrain," includes test runs where the systems predominantly navigated a gravel trail, but also encountered sections of unstructured forest terrain. In contrast, the "Forest" category is more exclusive. It comprises runs that were executed solely within unstructured forest terrain.

In the secondary analysis, attention shifts to two specific test runs, both characterized by extended distances and durations compare to the other tests. The first of these encompasses a long-distance run set exclusively within a unstructured forest terrain. The second test presents a mixed terrain: it begins on a gravel trail while the end point is situated in a unstructured forest terrain. Here the main caveat is that the gravel trail could be followed for various amount of time, and still be a viable route in order to reach the end point.

5.2 Energy efficiency

Energy values, which represent the total energy expended from the starting point A to the end point B of a specific experiment, are presented in Joules. Given the variability in test durations, distances traversed, and the inherent characteristics of each terrain, the energy metric was normalized. This normalization ensures that the data can be analyzed in a comparable manner, allowing for more accurate insights into the efficiency of each mode across different terrains.

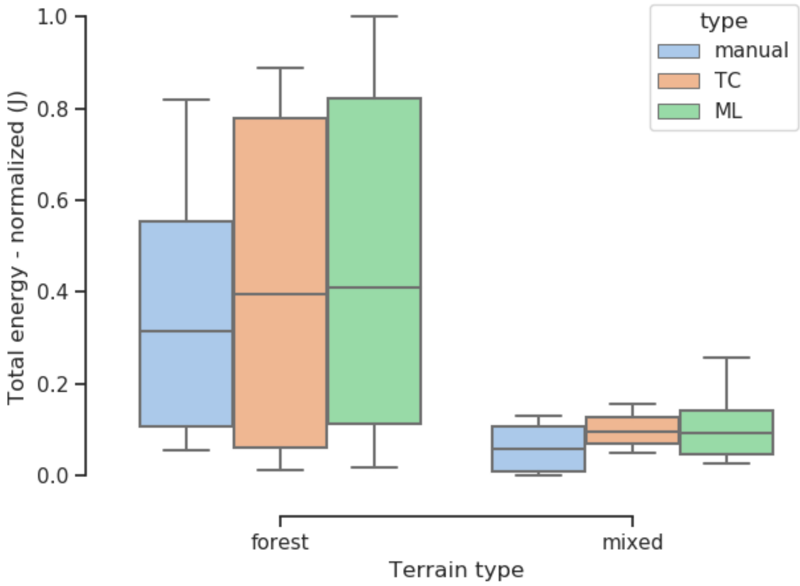


Figure 5.1: Normalized total energy consumption (J) per terrain type

The figure 5.1 of total energy consumption across the two terrain categories show a clear trend: the human operator consistently expends the least energy in the majority of test runs. Both of the autonomous systems tend to consume the same amount of energy on each terrain type.

5.2.1 Effects of steering behaviour

However, this result is not solely achieved by the human operator navigating the terrain more efficiently compared to the autonomous systems. The heatmap in figure 5.2 depicting accumulated yaw direction changes demonstrate that human operators make limited amount of steering adjustments compared to the autonomous systems. Given the robot's differential drive, any form of lateral movement might result in increased energy consumption, which can be attributed to the increased wheel slip and friction, which happen when making turning adjustments, as opposed to moving in a straight trajectory.

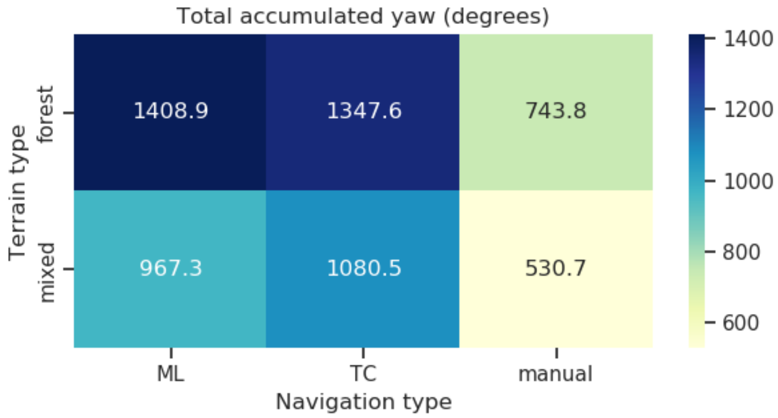


Figure 5.2: Accumulated yaw change per terrain type (degrees)

The human's minimalistic approach to steering is not necessarily a conscious strategy. Rather, it stems from the limited ability to access terrain down to more subtle terrain characteristics within it in real-time. Comparing the autonomous systems paints a similar picture. The ML-based system tends to make lateral adjustments more often than TC system, which, in turn, increases its energy consumption for the same test runs. Drawing from these observations, one can infer that energy consumption is to a large degree connected with the navigation habits exhibited by each mode.

When observing the human operator's inclination to mainly move straight without much adjustment to the terrain, an important question is if this seemingly linear approach would be more beneficial for autonomous systems as well. Or on the other hand, is it the autonomous systems that seem too proactive or even excessive in their steering adjustments, thereby leading to higher energy consumption? To provide more context to these nuances, a series of shorter, controlled tests were conducted that are in detail described in section 4.2.

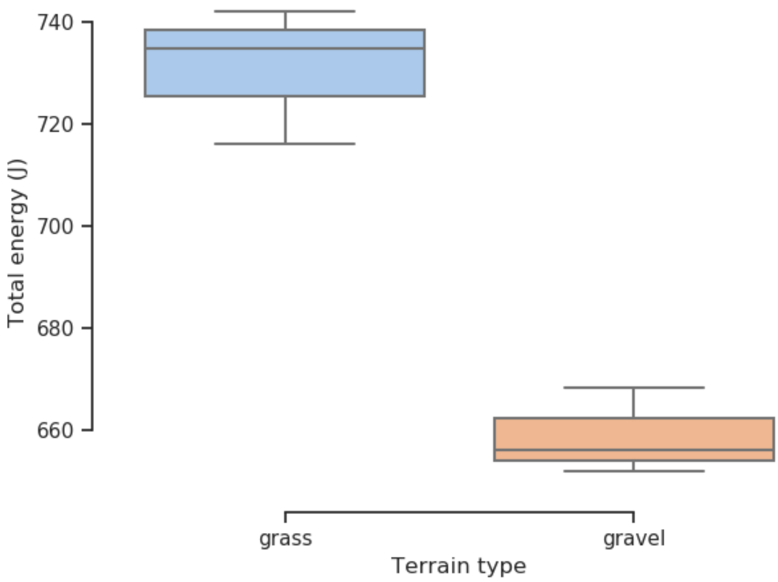


Figure 5.3: Total energy consumption (J) per terrain type in a controlled experiment number 1

The core objective of this experiment was to gauge the robot's ability to proprioceptively measure the variations between two terrains with distinct ground textures. Anticipating the outcomes based on the nature of the terrains, it was hypothesized that the gravel trail would yield a more energy-efficient traverse. The results from these tests in figure 5.3 confirm this hypothesis. The robot's traversal on the gravel trail indeed proved to be more energy-conserving than on the grassy terrain. Based on these results, it can be inferred that there are meaningful navigation decisions to be made when traversing non-linear terrain. However, the exact point of diminishing returns of improved energy efficiency when applying more steering effort is not known.

A second contextual experiment was done on a flat gravel trail, where one half of the driven stretch has negative obstacles the entire way, while the other half is completely flat. The human operator only went straight through the part with more undesirable terrain, while the two autonomous systems were sent off from the same starting point.The aim was twofold- not only to see if the autonomous systems would steer off towards the flatter part as intended, but more importantly, to see how the energy consumption changes while increasing steering effort towards a better terrain versus driving straight through a worse terrain profile.

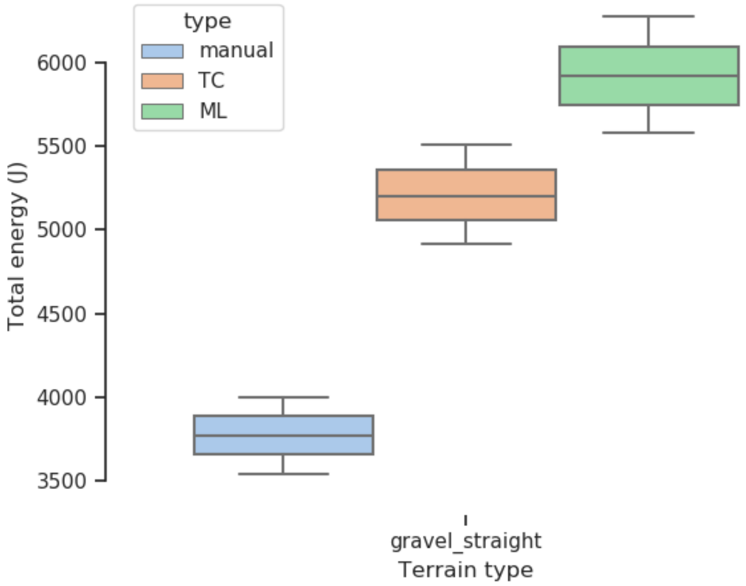


Figure 5.4: Total energy consumption (J) in a controlled experiment #2

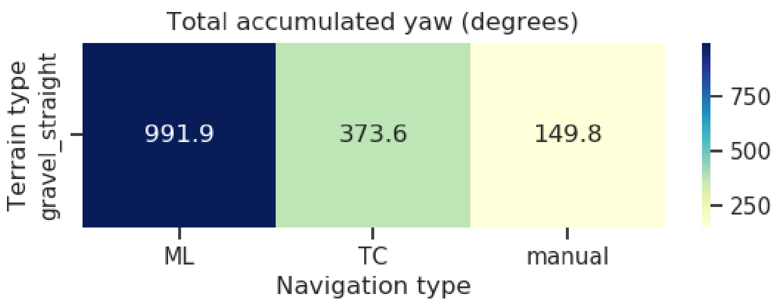


Figure 5.5: Accumulated yaw change per terrain type (degrees) in a controlled experiment #2

As can be seen in figure 5.4, TC system maintains similar energy efficiency per distance unit with a higher total yaw input (fig. 5.5) during the test's duration compared to the human operator. At first look, the energy efficiency of the TC system is matching the human operator, and does not bring any meaningful improvement, however it's important to note that this particular implementation tends to apply steering when it is not completely necessary. If a similar system could manage to handle very uniform areas such as in this example more consistently, it would follow that the energy efficiency would also improve further. Overall, this finding is a good indication that an autonomous system with local terrain awareness can make better navigational decisions when applicable by using more costly maneuvers and benefit from them.

The ML system on the other hand, while it did avoid the rough terrain, it also used excessive steering effort due the the implementation's insufficient capabilities in an exclusively flat terrain. This leads to higher energy consumption then what could be possible without this behaviour present, and therefore does not show the most realistic representation. This kind of behaviour was present during other traversals happening on similar gravel terrain, and is discussed in the next section.

The core insight from this experiment is the potential of an autonomous terrain characterization system, be it classical or machine-learning-based, to make navigational decisions resulting in reduced energy consumption compared to systems without such capabilities. Yet, it's imperative that the system is robust, equipped to handle scenarios that may cause abnormal steering. Specifically in this case, the TC system handled the scenario better compared to the ML system.

5.2.2 Energy consumption in long duration experiments (*Experiment F*)

Starting with the long duration test number one situated in a forest terrain (fig. 5.6), we can see the same trends as beforehand. The human operator uses the least amount of energy with the least amount of total yaw applied during the whole test duration (fig. 5.7). The two autonomous systems achieve similar results in this test. The ML system achieves similar results compared to the TC system. In this case, the total accumulated yaw for both systems is also similar.

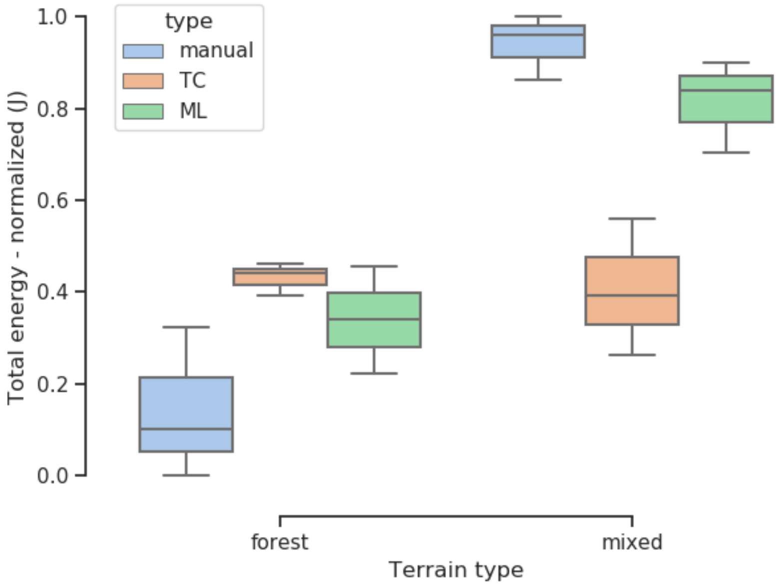


Figure 5.6: Normalized total energy consumption (J) per terrain type in long duration experiments

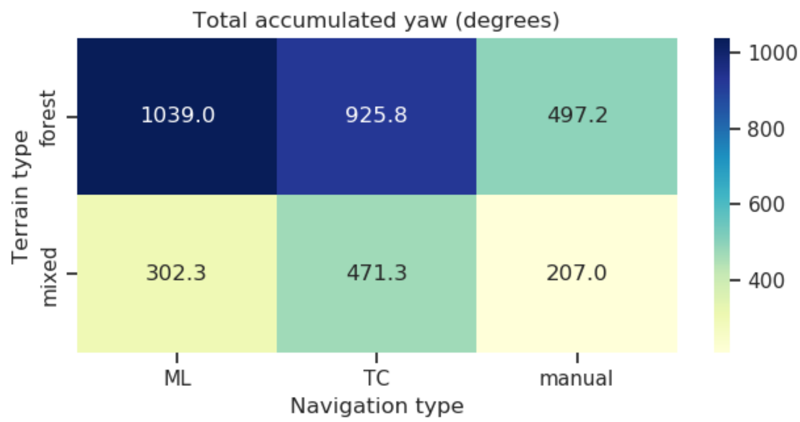


Figure 5.7: Accumulated yaw change per terrain type (degrees) in long duration experiments

Long duration run number two reveals one essential implication of using only local perception for the autonomous systems. The end point of this particular test was placed in a unstructured forest terrain, while the common starting point was on a gravel trail. However, it was a viable possibility to close the distance towards the end point by taking a longer overall path that mostly stays on the gravel trail.

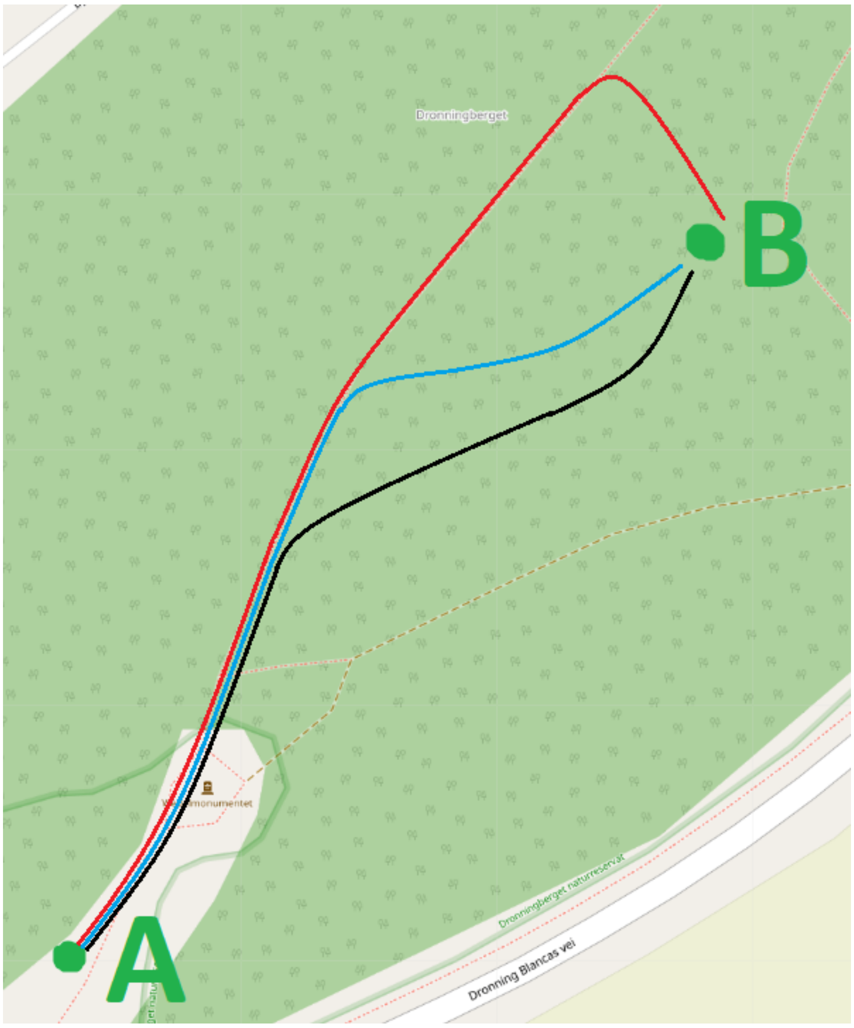


Figure 5.8: Approximation of paths taken in long duration run #2 (mixed terrain), based on RGB images. Red = human, Blue = TC system, Black = ML system

This was indeed executed this way by the human operator, as at first glance that was the most natural way to approach the scenario. The autonomous systems, however, constrained by their limited local perception, diverged from the gravel terrain sooner. As depicted in figure 5.6, this results in an overall higher amount of energy consumed for the human operator, but simultaneously having a lower energy per distance compared to the autonomous systems. While this result is contradicting the previous findings, it is an example of where global based decisions have important navigational implications. Such findings indicate the potential advantages of combining local terrain awareness with global navigational insights.

While the two autonomous systems solved this problem similarly, each system transitioned from gravel to forest terrain at different points. This might be attributed either to the navigation system in use, due to how the GPS component is implemented. However, as will be discussed in a separate section, the aforementioned discrepancy could also be the result of different object-detection behaviours of each system. The TC system completed the transition in an area with minor obstacles, while the ML system chose to navigate through a set of larger obstacles which could explain the higher energy usage by the ML system due to more navigational effort required.

5.2.3 Roughness

In evaluating the performance of the various navigation strategies, it's not just energy consumption that matters. We must consider other metrics that gauge the effectiveness of each system, especially in areas where the benefits may not be immediately apparent in terms of energy. One crucial measure in this context is the ability to navigate a robot towards safer terrains, which is fundamental to the overarching objectives of this technology. The 'roughness' metric offers an insight into this aspect. Defined as the summed average of standard deviation of pitch and roll (representing the robot's lateral movements in x and y directions), it gives us a quantifiable measurement of how smoothly the robot traverses varied terrains over the span of a test run.

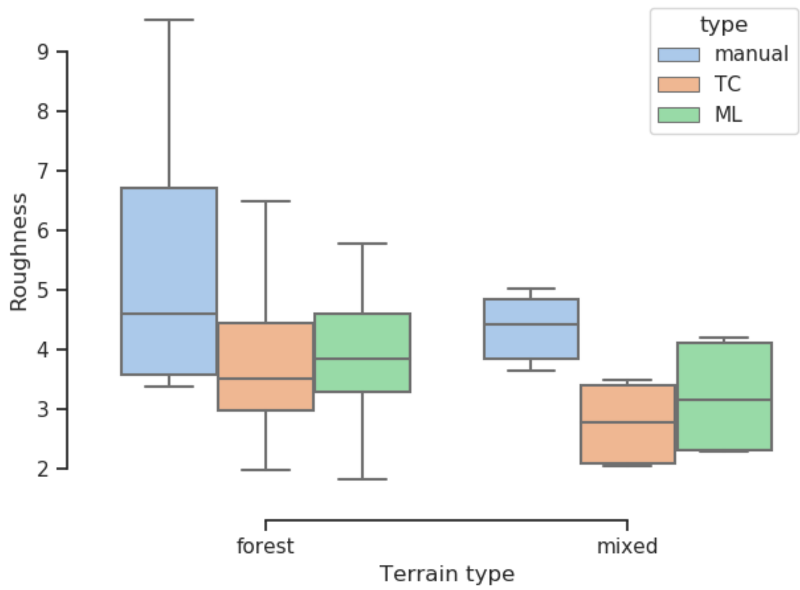


Figure 5.9: Roughness per terrain type

As seen in the figure 5.9, results gathered by the human operator across the two terrain types has consistently higher roughness than that of the two autonomous systems. As previously shown, human operator also inputs the least amount of yaw adjustments (fig. 5.2), meaning the robot is sent straight ahead through terrain more often. The TC system exhibits the lowest roughness levels, which confirms observations of its capability at detecting and beneficially navigating minor terrain variances. Meanwhile, the ML system is not as capable as the TC system due to the capabilities of this system and how it seems to operate in this implementation, which is described more in detail in the next section.

Overall, both autonomous systems achieve lower amounts of roughness that is subjected upon the robot, with the classical technique performing better compared to machine learning. This is an accomplishment in one of the two areas that would make autonomous terrain characterization a viable solution for remote robotic applications in the real world. This however depends on the what each application requires. Achieving this represents a milestone in one of the key areas that make terrain characterization a feasible solution for remote robotic applications in the real-world. Nevertheless, the applicability depends on the specific requirements of each individual application, where either energy consumption or lower roughness is preferred.

5.2.4 Roughness in long duration experiments (*Experiment F*)

In evaluating the long-duration tests from the perspective of roughness, similar patters can be observed due to the use of local perception (fig. 5.10). On a longer duration run exclusively in unstructured forest terrain, the roughness achieved by a human operator is higher compared to both autonomous systems. The roughness achieved by an ML system is slightly higher compared to the TC system.

In the same manner as previously, the results for the second long duration on a mixed terrain run are somewhat counter-intuitive. It could be expected that the autonomous systems would achieve lower roughness as it generally does, but in this case it does not, as the human operator took a much longer path along the gravel trail, which naturally results in lower roughness measurements for that particular run. This kind of a solution to this problem again highlights the needs for a global perspective added to the functionalities of a local terrain perception.

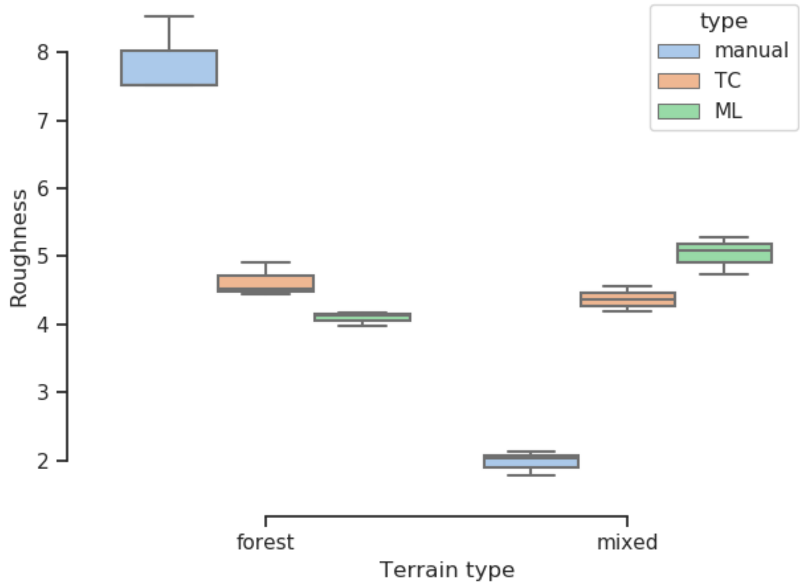


Figure 5.10: Roughness per terrain type in long duration experiments

From these findings, a broader conclusion can be drawn: in robotic navigation, there's an intricate balance between energy consumption and experienced roughness. This balance might be adjusted either way for a potential remote robotic mission, depending on a number of factors. For instance, in a robotic mission, considerations such as energy reserves, the robot's traversal capabilities, and the comprehensive understanding of the environment will dictate whether the focus should lean towards energy efficiency or smoother traversal of the terrain.

5.3 Traversal from point A to B

Several tests of shorter durations and distances were done in order to quantify and detect the behaviour and capabilities of the autonomous systems when put in edge case scenarios, as well as finding qualitative comparisons between the two autonomous systems. The performed experiments were aimed at finding out if the systems are capable of handling certain situations, as well as which areas of the tested field proved to be the most challenging. All of the experiments used to come the following conclusions were executed according to their descriptions in section 4.2.

The first tests of *Experiment B* were intended to show how transitions between contrasting terrains are handled. Both autonomous systems managed go traverse from point A to point B successfully. Both have made the transitions between the two types of terrain in a reasonable place. There were not any noticeable differences between the TC and ML systems in this specific scenario.

During *Experiment C*, where the systems were tasked to track and eventually cross a terrain transition, both managed to reach the point B, identifying feasible crossing points. However, their navigation differed notably at a curving terrain transition alongside a gravel trail. The ML system adhered close to the transition line, while the TC system exhibited more oscillatory movement, frequently deviating between the transition and an excessive heading offset away from point B.

One test of *Experiment D* was done to test a typical terrain scenario that can often be encountered in a forest environment. The aim was to follow a clearly discernible trail through thick vegetation as long as possible without steering off of the trail. Both TC and ML systems were able to reach the the same final distance away from point A. The TC system performed better and kept within the boundaries more steadily, and never hitting any of the nearby non-desirable terrain. In contrast, ML displayed a noticeable drift towards the right side of the trail, indicative of its heavy dependence on discernible and contrasting edges in its input images. This inherent bias not only made its navigation less precise but also frequently steer the robot into more challenging terrain. Such behaviors highlight the ML system's challenges when presented with paths that require subtle and nuanced navigation decisions. While it possesses the capacity to recognize and maneuver around major obstacles, it appears to struggle in scenarios where maintaining a delicate course is paramount.

The test runs of *Experiment E* were executed only in an unstructured forest terrain, with both point A and point B situated in such a terrain. The main goal is to gather data from this specific terrain, but also to observe how the autonomous systems perform with regards to navigating around unstructured terrain. Similarly in these tests, both systems managed to reach the end point without getting stuck. Biggest differences between TC and ML systems was that they reacted to various obstacles differently, which is documented in more detail in the next subsection.

5.3.1 Obstacle avoidance and terrain navigation

There were different capabilities and behaviours observed with regards to obstacle avoidance and maneuvering through the terrain variations encountered in the traversed areas. When compared to a human operator, that had a wider and longer (more global) vision of the surroundings, the two implemented systems operated differently. Instead, the automated systems are limited to a narrow field of view, and do not have sufficient vision range to be able to make decisions ahead of time. These are the concise observations with regards to the topic at hand:

1. ML system and high contrast environments:
   - Nature of input: The ML system's reliance on sharp contrast, especially in areas with large height variations, proved beneficial in some contexts. The heightened contrast in the input heightmap, likely a consequence of the dataset's adjusted edge detection, ensures effective navigation in terrains with sharp-edged obstacles, such as trees and highly contrasting terrain transitions.
   - Strengths: In areas of an unstructured forest terrain which feature pronounced height variations due to trees, rocks and terrain transitions, this approach capitalizes on the model's trained inclination to detect sharp edges.
   - Weaknesses: However, in more homogenous environments like flat gravel terrains, this edge-detection prowess becomes a liability. The absence of significant variation and pronounced edges, combined with noisy sensor data and a navigation algorithm not tailored to handle such noise, results in erratic steering adjustments. This is most evident in the yaw accumulation heatmap.
2. TC system and real-time calculations:
   - Nature of input: Operating on real-time raw data, the TC system's proficiency lies in distinguishing minor terrain variations, offering a more granular and nuanced navigational action.
   - Strengths: Not only can the TC discern when a favorable path is present, but its real advantage is in situations where optimal paths are not immediately

evident to human observers.

- Weaknesses: However, while the TC system seems well-equipped to handle the subtleties of varying terrains, it might be prone to over-analysis in certain scenarios, leading to possible inefficiencies or over-caution.

5.3.2 Exteroception vs. proprioception

Throughout various runs across different scenarios, distinct conclusions can be drawn regarding the two sensory approaches for robotic perception in terrain characterization: exteroceptive and proprioceptive. The implication of each approach are extensively debated in prior terrain characterization research. This is due to the different applicability of both approaches which might be more suitable in different situations. In applications that are highly susceptible to wheel slip, such as interplanetary robotic applications, proprioceptive techniques often prove more reliable given their direct sensor-to-terrain interaction. On the other hand, research has often employed the usage of exteroceptive techniques in similar application as in this project, or in a widely used conventional in-door obstacle avoidance.

In this project, the TC system is purely exteroceptive. While the ML system is exteroceptive during operation, it is based on proprioceptive measurements during the training procedures. The results indicate that a purely exteroceptive approach excels in contexts where terrain information is largely available in advance through the available means. The decision making based on high resolution raw vision data has the capabilities of discerning the minor variations in terrain in a local vicinity, which allows a robot to act upon it. The shortcomings surface in situations when a terrain cannot be characterized by vision only, which can lead to navigational actions that are suboptimal.

On the other hand, it can be argued that a technique that combines both a proprioceptive and exteroceptive approach into a one unified terrain characterization system, could be able to make more nuanced decisions. Such a system would not only rely on real-time visual data, but would also draw correlations from prior physical interactions with similar terrain characteristics. This is the capability a machine learning based system has over a classical one, but requires much more complex implementation mainly driven by a robust and accurately measured datasets.

5.4 Limitations

In the course of this research, several key limitations were identified. Understanding these limitations is essential as they offer direction for future refinements and research avenues in the realm of autonomous terrain navigation.

The usage of low-cost, widely accessible sensors, might have confined the quality and breadth of terrain data available that was captured. This choice, while practical, potentially overlooks nuances that more sophisticated sensors could capture. These limitation became evident with regards to the quality of the collected dataset that was used to train the machine learning model. While the network produced a functional model, the employed dataset was limited in volume compared to what turned out to necessitate a more robust solution.

Another significant limitation was the short-term decision-making approach adopted by both classical and machine learning techniques. Relying heavily on immediate visual cues could at times overshadow the benefits of a broader, more holistic terrain perspective. While it is a direct consequence of the sensors being used, the navigation system limits the capabilities of how the system is able to act upon gained terrain information. This limitation mainly stems from the simplicity of the navigation system and usage of constant angular and linear velocity for mobility.

Lastly, the specific design of our experiments raises questions about the universality of our findings. Given the diversity of conditions and terrains a robotic system might face, it's important to critique how applicable the results are in a more general use case.

5.5 Future Work

Following the identified limitations of this study, this thesis outlines avenues for future research:

Firstly, refining and expanding the dataset is paramount. As we discovered, the need for a more expansive dataset is crucial in development of a more robust machine learning based technique for terrain characterization. The dataset must cover the vast dimensionality of an environment such as the one covered during the experimentation phase, while providing enough resolution in both the visual and numerical data such that a machine learning model can effectively discern both the subtle and pronounced terrain features.

The development of a more sophisticated navigation system is also needed. A system that can adjust its angular and linear velocity in response to terrain demands can offer a more effective navigational approach. This capability coupled with a global vision overview would reduce system's dependence on short-term navigational decisions that are based solely on the immediate visual cues.

By addressing these areas, we can better position autonomous robot navigation for success in unpredictable and unstructured environments in a wider range of applications.

Chapter 6

Conclusion

In concluding this research, the project sought to explore how a deep-learning-based terrain characterization measures up against a classical analytical approach when integrated into an autonomous system designed for navigation in an unstructured environment. While the system's physical abilities could handle these conditions, its navigational system needs the required function to the successfully operate in a remote setting. The overarching ambition of this technology is to eventually achieve full autonomy in in environments that have predominantly been off-limits to non-manually operated robots. This has countless applications in areas where a human's presence is in danger and has to be accessed by a machine. This project, therefore, set out to to explore what's needed to develop a system that's capable of achieving such future.

This project included all the necessary aspects that are needed in transitioning a technological concept into a tangible solution. Starting with a robotic platform, it was equipped with both the essential hardware and software, enabling it to utilize two terrain characterization techniques using a common navigation system. The implemented sensor suite consists of widely accessible sensors. Leveraging a stereo camera, IMU, GPS module, and wheel odometers, two early prototypes of the studied techniques were implemented. These were then rigorously tested through in-field experiments specifically designed to provide more information about the project's research question.

A total of seven main categories of experiments were outlined specifically to find an answer to how the implemented techniques perform in the real-world. These experiments shed light on specific behaviors, identified constraints, and provided a basis for a comparative analysis of the two approaches. The main observation was that the autonomous systems tended to consume more energy than compared to a human operator, primarily due to the increased navigational efforts executed by the systems. However, it could be inferred from the results that there is a balance to be found between an over-cautious navigation and energy efficient maneuvers that provides the most optimal solution to terrain characterization.

While the energy consumption was overall higher than what was achieved by a human, the autonomous systems showed an ability to optimize the second important function of terrain characterization. Both systems consistently recorded lower measure of experienced roughness, indicating that the systems were able to navigate terrain in a safer manner. This functionality, albeit more energy-intensive, underlines the priority of minimizing risks in more hazardous unstructured environments.

Additionally, both implemented systems showcased distinct ways of interpreting obstacles and terrain features, stemming from their foundational processes. The classical technique excelled in its sensitivity to minor terrain variations, benefiting from its reliance on high-resolution visual data. In contrast, the machine learning-based approach leaned heavily on recognizing pronounced features within its input imagery, likely a reflection of its training, where emphasis was placed to highlight excessive terrain transitions. Striking a balance between these could lead to a comprehensive system adept at discerning both minor and major terrain features simultaneously.

The complexity of unstructured terrain introduces a multitude of variables and unique elements that pose learning challenges. The process of dataset collection carried out during this project, essential for training the machine learning model, culminated in several valuable outcomes. It became apparent that training of a deep-learning network requires a much larger dataset than previously anticipated, to accomplish a comprehensive model of a real-world environment. This accentuates the need for an datasets with terrain imagery combined with highly accurate proprioceptive measurements when applicable.

The adaptability of a classical method such as the TC system is advantageous in the short term context, as it can easily be employed in a brand new environment and could provide satisfactory results. The shortcoming of such a system comes from the fact that further expansion or improvements become steadily more complex and challenging to implement, and might only provide diminishing improvements. In contrast, a machine learning technique is more complex to get going, but with more effort spent on data collection, in the long term it has the potential to improve past the limits of a classical approach. Even with a limited dataset, leveraging relatively straightforward neural network and inadequate sensors, we were able to show positive results of a machine learning based system that were not far behind the capabilities of a classical approach. Although the outcomes were not always optimal when compared to a conventional remote operation of robotic systems, both classical and machine learning techniques managed to navigate the robot successfully from point A to point B. This current iteration is not universally adaptable in it's current stage however, and extensive refinements are necessary for a broader use case.

The conclusions drawn from this research underscore that the journey towards perfecting terrain characterization for robotic applications remains ongoing. It became apparent that both classical and machine learning techniques could significantly benefit in a number of areas. For instance, a wider global perspective of the surroundings would aid in making decisions that are more optimal over a longer term. However, some of these challenges could be attributed to the rudimentary navigation system that was used in this implementation. Additionally, the previously mentioned dataset collection challenges remain as the main hurdle towards a complete solution that is based on any machine learning-based technique.

In the rapidly evolving field of robotic systems and autonomous navigation, the exploration of both classical and machine learning-based terrain characterization approaches opens new horizons for safe and efficient mobility in uncharted terrains. This research serves as a foundational study, highlighting the strengths and limitations of both methodologies in real-world scenarios. The journey towards perfecting this concept continues, with the hope that the findings of this thesis light the path for future research in the realm of autonomous terrain navigation.

Bibliography

Angelova, A., Matthies, L., Helmick, D., & Perona, P. (2007). Learning and prediction of slip from visual information. *Journal of Field Robotics*, *24*(3), 205–231. https://doi.org/10.1002/rob.20179

Bekker, M. G. (1969a). *Introduction to terrain-vehicle systems*. University of Michigan Press. Retrieved June 7, 2022, from http://books.google.com/books?id=nNg8AAAAIAAJ

Bekker, M. G. (1969b). *Introduction to Terrain-vehicle Systems*. University of Michigan Press.

Borges, P., Peynot, T., Liang, S., Arain, B., Wildie, M., Minareci, M., Lichman, S., Samvedi, G., Sa, I., Hudson, N., Milford, M., Moghadam, P., & Corke, P. (2022). A Survey on Terrain Traversability Analysis for Autonomous Ground Vehicles: Methods, Sensors, and Challenges. *Field Robotics*, *2*(1), 1567–1627. https://doi.org/10.55417/fr.2022049

Broggi, A., Caraffi, C., Fedriga, R., & Grisleri, P. (2005). Obstacle Detection with Stereo Vision for Off-Road Vehicle Navigation. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, 65–65. https://doi.org/10.1109/CVPR.2005.503

Carlucho, I., De Paula, M., Villar, S. A., & Acosta, G. G. (2017). Incremental Q-learning strategy for adaptive PID control of mobile robots. *Expert Systems with Applications*, *80*, 183–199. https://doi.org/10.1016/j.eswa.2017.03.002

Collins, J., Howard, D., & Leitner, J. (2019). Quantifying the Reality Gap in Robotic Manipulation Tasks. *2019 International Conference on Robotics and Automation (ICRA)*, 6706–6712. https://doi.org/10.1109/ICRA.2019.8793591

Dallas, J., Cole, M. P., Jayakumar, P., & Ersal, T. (2020, March 3). *Neural network based terramechanics modeling and estimation for deformable terrains* (arXiv:2003.02635). arXiv. Retrieved June 10, 2022, from http://arxiv.org/abs/2003.02635

DARPA. (2022). *RACER Revs Up for Checkered Flag Goal of High-Speed, Off-Road Autonomy*. Retrieved June 9, 2022, from https://www.darpa.mil/news-events/2022-01-13

Dumitrascu, B., Filipescu, A., Minzu, V., & Filipescu, A. (2011). Backstepping control of wheeled mobile robots. *15th International Conference on System Theory, Control and Computing*, 1–6.

ELROB. (2022). *Robotic Competition - European Land-Robot Trial 2022 | ELROB.org*. Retrieved June 9, 2022, from https://www.elrob.org/elrob-2022

Espinoza, A., Dar, T., & Longoria, R. G. (2021). Estimating vehicle-terrain interaction parameters from tracked-robot sensor data. *IFAC-PapersOnLine*, *54*(20), 644–649. https://doi.org/10.1016/j.ifacol.2021.11.244

Fareh, R., Saad, M., Khadraoui, S., & Rabie, T. (2016). Lyapunov-Based Tracking Control for Nonholonomic Wheeled Mobile Robot. https://www.researchgate.net/publication/319555601_Lyapunov-Based_Tracking_Control_for_Nonholonomic_Wheeled_Mobile_Robot

Farias, G., Garcia, G., Montenegro, G., Fabregas, E., Dormido-Canto, S., & Dormido, S. (2020). Position control of a mobile robot using reinforcement learning. *IFAC-PapersOnLine*, *53*(2), 17393–17398. https://doi.org/10.1016/j.ifacol.2020.12.2093

Filitchkin, P., & Byl, K. (2012). Feature-based terrain classification for LittleDog. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1387–1392. https://doi.org/10.1109/IROS.2012.6386042

Findeisen, R., & Allgöwer, F. (2002). An Introduction to Nonlinear Model Predictive Control.

Galati, R., & Reina, G. (2019). Terrain Awareness Using a Tracked Skid-Steering Vehicle With Passive Independent Suspensions. *Frontiers in Robotics and AI*, *6*. Retrieved April 13, 2022, from https://www.frontiersin.org/article/10.3389/frobt.2019.00046

Helmick, D., Angelova, A., & Matthies, L. (2009). Terrain Adaptive Navigation for planetary rovers. *Journal of Field Robotics*, *26*(4), 391–410. https://doi.org/10.1002/rob.20292

Howard, A., & Seraji, H. (2001). Vision-based terrain characterization and traversability assessment. *Journal of Robotic Systems*, *18*(10), 577–587. https://doi.org/10.1002/rob.1046

Hu, Y., Su, H., Zhang, L., Miao, S., Chen, G., & Knoll, A. (2019). Nonlinear Model Predictive Control for Mobile Robot Using Varying-Parameter Convergent Differential Neural Network. *Robotics*, *8*(3), 64. https://doi.org/10.3390/robotics8030064

Iagnemma, K., & Dubowsky, S. (2004, July 8). *Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers*. Springer Science & Business Media.

IEEE. (2011). *Husky - ROBOTS: Your Guide to the World of Robotics*. Retrieved June 9, 2022, from https://robots.ieee.org/robots/husky/

Ishigami, G., Miwa, A., Nagatani, K., & Yoshida, K. (2006). Terramechanics-based analysis on slope traversability for a planetary exploration rover.

Jakobi, N., Husbands, P., & Harvey, I. (1995). Noise and the reality gap: The use of simulation in evolutionary robotics. In *Advances in Artificial Life* (pp. 704–720). Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-59496-5_337

Khan, Y. N., Komma, P., & Zell, A. (2011). High resolution visual terrain classification for outdoor robots. *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 1014–1021. https://doi.org/10.1109/ICCVW.2011.6130362

Kolmanovsky, I., & McClamroch, N. (1995). Developments in nonholonomic control problems. *IEEE Control Systems Magazine*, *15*(6), 20–36. https://doi.org/10.1109/37.476384

Larson, J., Trivedi, M., & Bruch, M. (2011, January 1). *Off-Road Terrain Traversability Analysis and Hazard Avoidance for UGVs*. CALIFORNIA UNIV SAN DIEGO DEPT OF ELECTRICAL ENGINEERING. Retrieved June 9, 2022, from https://apps.dtic.mil/sti/citations/ADA536652

Li, C., & Lewis, K. (2022). The Need for and Feasibility of Alternative Ground Robots to Traverse Sandy and Rocky Extraterrestrial Terrain. *Advanced Intelligent Systems*, *n/a*(n/a), 2100195. https://doi.org/10.1002/aisy.202100195

Lim, H., Kang, Y., Kim, C., Kim, J., & You, B.-J. (2008). Nonlinear Model Predictive Controller Design with Obstacle Avoidance for a Mobile Robot. *2008 IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications*, 494–499. https://doi.org/10.1109/MESA.2008.4735699

Lin, P., Guarini, M., Abney, K., Bekey, G. A., Allen, C., Wallach, W., Hughes, J. J., Bringsjord, S., Taylor, J., & Sharkey, N. (2011). *Robot Ethics: The Ethical and Social Implications of Robotics*. MIT Press. Retrieved June 9, 2022, from http://ebookcentral.proquest.com/lib/hioa/detail.action?docID=3339351

Liu, T., Liu, D., Yang, Y., & Chen, Z. (2019). Lidar-based Traversable Region Detection in Off-road Environment. *2019 Chinese Control Conference (CCC)*, 4548–4553. https://doi.org/10.23919/ChiCC.2019.8865250

Lopez-Arreguin, A. J. R., & Montenegro, S. (2021). Machine learning in planetary rovers: A survey of learning versus classical estimation methods in terramechanics for in situ exploration. *Journal of Terramechanics*, *97*, 1–17. https://doi.org/10.1016/j.jterra.2021.04.005

Lourenco, D. S. B., Ferreira, J. F., & Portugal, D. (2020). 3D Local Planning for a Forestry UGV based on Terrain Gradient and Mechanical Effort.

Maeda, M., Maeda, Y., & Murakami, S. (1991). Fuzzy drive control of an autonomous mobile robot. *Fuzzy Sets and Systems*, *39*(2), 195–204. https://doi.org/10.1016/0165-0114(91)90212-9

Malu, S. K., & Majumdar, J. (2014). Kinematics, Localization and Control of Differential Drive Mobile Robot. *Global Journal of Research In Engineering*. Retrieved June 6, 2022, from https://engineeringresearch.org/index.php/GJRE/article/view/1233

Martin, S. C. (2018). *Proprioceptive Sensing of Traversability for Long-Term Navigation of Robots* (UNSPECIFIED). Queensland University of Technology. https://doi.org/10.5204/thesis.eprints.117074

Nagatani, K., Kiribayashi, S., Okada, Y., Otake, K., Yoshida, K., Tadokoro, S., Nishimura, T., Yoshida, T., Koyanagi, E., Fukushima, M., & Kawatsuma, S.

(2013). Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. *Journal of Field Robotics*, *30*(1), 44–63. https://doi.org/10.1002/rob.21439

Nascimento, T. P., Dórea, C. E. T., & Gonçalves, L. M. G. (2018). Nonholonomic mobile robots' trajectory tracking model predictive control: A survey. *Robotica*, *36*(5), 676–696. https://doi.org/10.1017/S0263574717000637

Nygaard, T. F., Martin, C. P., Torresen, J., Glette, K., & Howard, D. (2021). Real-world embodied AI through a morphologically adaptive quadruped robot. *Nature Machine Intelligence*, *3*(5), 410–419.

Ojeda, L., Borenstein, J., Witus, G., & Karlsen, R. (2006). Terrain characterization and classification with a mobile robot. *Journal of Field Robotics*, *23*(2), 103–122. https://doi.org/10.1002/rob.20113

Ordonez, C., Alicea, R., Rothrock, B., Ladyko, K., Nash, J., Thakker, R., Daftry, S., Harper, M., Collins, E., & Matthies, L. (2020). Characterization and Traversal of Pliable Vegetation for Robot Navigation. In J. Xiao, T. Kröger & O. Khatib (Eds.), *Proceedings of the 2018 International Symposium on Experimental Robotics* (pp. 293–304). Springer International Publishing. Retrieved April 13, 2022, from http://link.springer.com/10.1007/978-3-030-33950-0_26

Ostafew, C. J., Schoellig, A. P., Barfoot, T. D., & Collier, J. (2016). Learning-based Nonlinear Model Predictive Control to Improve Vision-based Mobile Robot Path Tracking. *Journal of Field Robotics*, *33*(1), 133–152. https://doi.org/10.1002/rob.21587

Papadakis, P. (2013). Terrain traversability analysis methods for unmanned ground vehicles: A survey. *Engineering Applications of Artificial Intelligence*, *26*(4), 1373–1385. https://doi.org/10.1016/j.engappai.2013.01.006

Park, H., Sun, J., Pekarek, S., Stone, P., Opila, D., Meyer, R., Kolmanovsky, I., & DeCarlo, R. (2015). Real-Time Model Predictive Control for Shipboard Power Management Using the IPA-SQP Approach. *IEEE Transactions on Control Systems Technology*, *23*(6), 2129–2143. https://doi.org/10.1109/TCST.2015.2402233

Prabhu, S. M., & Garg, D. P. (1998). Fuzzy-logic-based Reinforcement Learning of Admittance Control for Automated Robotic Manufacturing. *Engineering*

Applications of Artificial Intelligence, *11*(1), 7–23. https://doi.org/10.1016/S0952-1976(97)00057-2

Reddy, S. K., & Pal, P. K. (2016). Detection of traversable region around a mobile robot by computing terrain unevenness from the range data of a 3D laser scanner. *International Journal of Intelligent Unmanned Systems*, *4*(2), 107–128. https://doi.org/10.1108/IJIUS-08-2015-0009

Rezaee, A. (2017). Model predictive Controller for Mobile Robot. *Transactions on Environment and Electrical Engineering*, *2*(2), 18–23. https://doi.org/10.22149/teee.v2i2.96

Robotnik. (2023, February 28). *Robots: Warehouse management solutions. | Robotnik ®*. Robotnik. Retrieved August 15, 2023, from https://robotnik.eu/benefits-of-mobile-robots-in-a-warehouse/

Rubio, F., Valero, F., & Llopis-Albert, C. (2019). A review of mobile robots: Concepts, methods, theoretical framework, and applications. *International Journal of Advanced Robotic Systems*, *16*(2), 1729881419839596. https://doi.org/10.1177/1729881419839596

Sathyamoorthy, A. J., Weerakoon, K., Guan, T., Liang, J., & Manocha, D. (2022, March 11). *TerraPN: Unstructured Terrain Navigation using Online Self-Supervised Learning*. Retrieved April 13, 2022, from http://arxiv.org/abs/2202.12873

Seraji, H., & Howard, A. (2002). Behavior-based robot navigation on challenging terrain: A fuzzy logic approach. *IEEE Transactions on Robotics and Automation*, *18*(3), 308–321. https://doi.org/10.1109/TRA.2002.1019461

Shijin, C. S., & Udayakumar, K. (2017). Speed control of wheeled mobile robots using PID with dynamic and kinematic modelling. *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 1–7. https://doi.org/10.1109/ICIIECS.2017.8275962

Siegwart, R., Nourbakhsh, I. R., Scaramuzza, D., & Arkin, R. C. (2011). *Introduction to Autonomous Mobile Robots, Second Edition*. MIT Press. Retrieved May 27, 2022, from http://ebookcentral.proquest.com/lib/hioa/detail.action?docID=3339191

Siva, S., Wigness, M., Rogers, J. G., & Zhang, H. (2021, January 1). *Robot Adaptation for Generating Consistent Navigational Behaviors over Unstructured Off-Road Terrain*. Retrieved April 13, 2022, from http://arxiv.org/abs/2101.00290

Sock, J., Kim, J., Min, J., & Kwak, K. (2016). Probabilistic traversability map generation using 3D-LIDAR and camera. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 5631–5637. https://doi.org/10.1109/ICRA.2016.7487782

Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2020, March 30). *Robot Modeling and Control*. John Wiley & Sons.

Stewart, C. N. (2011). Research ethics for scientists : A companion for students. *Wiley-Blackwell*.

Talukder, A., Manduchi, R., Castano, R., Owens, K., Matthies, L., Castano, A., & Hogg, R. (2002). Autonomous terrain characterisation and modelling for dynamic control of unmanned vehicles. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, *1*, 708–713 vol.1. https://doi.org/10.1109/IRDS.2002.1041474

Tzafestas, S. G. (2014a, January 1). 1 - Mobile Robots: General Concepts. In S. G. Tzafestas (Ed.), *Introduction to Mobile Robot Control* (pp. 1–29). Elsevier. Retrieved May 6, 2022, from https://www.sciencedirect.com/science/article/pii/B9780124170490000018

Tzafestas, S. G. (2014b, January 1). 5 - Mobile Robot Control I: The Lyapunov-Based Method. In S. G. Tzafestas (Ed.), *Introduction to Mobile Robot Control* (pp. 137–183). Elsevier. Retrieved May 6, 2022, from https://www.sciencedirect.com/science/article/pii/B9780124170490000055

Tzafestas, S. G. (2014c, January 1). 8 - Mobile Robot Control IV: Fuzzy and Neural Methods. In S. G. Tzafestas (Ed.), *Introduction to Mobile Robot Control* (pp. 269–317). Elsevier. https://doi.org/10.1016/B978-0-12-417049-0.00008-0

Vaidyanathan, S., & Azar, A. T. (2021, January 1). Chapter 1 - An introduction to backstepping control. In S. Vaidyanathan & A. T. Azar (Eds.), *Backstepping Control of Nonlinear Dynamical Systems* (pp. 1–32). Academic Press. Retrieved June 6, 2022, from https://www.sciencedirect.com/science/article/pii/B9780128175828000088

Vulpi, F., Milella, A., Marani, R., & Reina, G. (2021). Recurrent and convolutional neural networks for deep terrain classification by autonomous robots. *Journal of Terramechanics*, *96*, 119–131. https://doi.org/10.1016/j.jterra.2020.12.002

Weerakoon, K., Sathyamoorthy, A. J., Patel, U., & Manocha, D. (2022, March 3). *TERP: Reliable Planning in Uneven Outdoor Environments using Deep Reinforcement Learning*. Retrieved April 13, 2022, from http://arxiv.org/abs/2109.05120

Zhou, L., Wang, J., Lin, S., & Chen, Z. (2022). Terrain Traversability Mapping Based on LiDAR and Camera Fusion. *2022 8th International Conference on Automation, Robotics and Applications (ICARA)*, 217–222. https://doi.org/10.1109/ICARA55094.2022.9738526

Zidani, Gh., Drid, S., Chrifi-Alaoui, L., Benmakhlouf, A., & Chaouch, S. (2015). Backstepping controller for a wheeled mobile robot. *2015 4th International Conference on Systems and Control (ICSC)*, 443–448. https://doi.org/10.1109/ICoSC.2015.7153286