UNIVERSITY OF CALIFORNIA, SAN DIEGO

Off-road Obstacle Classification and Traversability Analysis in the Presence of Negative Obstacles

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Engineering Sciences (Electrical Engineering)

by

Jacoby Larson

Committee in charge:

Professor Mohan Trivedi, Chair
Professor Kenneth Kreutz-Delgado
Professor William Hodgkiss

2011

The thesis of Jacoby Larson is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____
Chair

University of California, San Diego

2011

DEDICATION

To my dear and precious wife,
who encourages me to do better and blesses my life,
and just understands me the way no one else does
To my energetic and sweet boys,
who can't wait to play with me when I come home,
and fill my days with fun and laughter

TABLE OF CONTENTS

Signature Page . . . iii

Dedication . . . iv

Table of Contents . . . v

List of Figures . . . vii

List of Tables . . . x

Acknowledgements . . . xi

Abstract of the Thesis . . . xii

Chapter 1 Introduction . . . 1
1.1 Research Goals, Motivation, and Scope . . . 1
1.2 Thesis Outline and Contributions . . . 4

Chapter 2 Related Research . . . 5
2.1 Video-based vs Lidar-based . . . 5
2.2 Military vs Consumer Systems . . . 6
2.3 Off-road vs On-road . . . 6
2.4 Mapping . . . 7
2.5 Classification . . . 8
2.6 Traversability . . . 9
2.7 Robotic System Comparison . . . 9

Chapter 3 Approach . . . 11
3.1 Data Capture . . . 12
3.2 Pre-processing (Representation of the Point Cloud) . . . 14
3.3 Negative Obstacles . . . 19
3.3.1 Detection Range . . . 20
3.3.2 NODR Classification Approach . . . 21
3.3.3 SVM Classification Approach . . . 27
3.3.4 Real Negative Obstacle Classification . . . 27
3.3.5 Translation to World Map . . . 29
3.4 Traversability Analysis . . . 30
3.5 Path Planning . . . 31

Chapter 4 Experimental Evaluation . . . 35
4.0.1 How to Measure Performance . . . 35
4.0.2 Setup of Simulated Environment . . . 36
4.0.3 Selecting of Test Cases . . . 36

4.0.4 Results of the Small Lidar Experiment (NODR Only) 38
4.0.5 Results of the Small Lidar Experiment (NODR and SVM) 40
4.0.6 Results of the Large Lidar Experiment (NODR Only) 40
4.0.7 Results of Large Lidar Experiment (NODR and SVM) 43
4.0.8 Results of Real Off-road Course 44

Chapter 5 Conclusion and Future Work 48

Bibliography 50

LIST OF FIGURES

Figure 1.1: "Soda-straw" video feed from a small ground robot 2
Figure 1.2: Navy reservists driving small UGV (in background) 3
Figure 1.3: Urban traffic intersection: image (left) and point cloud (right) 3

Figure 3.1: Method for Autonomy 12
Figure 3.2: Stopping Distance vs Speed 13
Figure 3.3: Max ATV UGV test-bed platform 15
Figure 3.4: Velodyne HDL-64E lidar sensor 15
Figure 3.5: iRobot Packbot UGV test-bed platform with mounted Nodding Hokuyo sensor and color camera 15
Figure 3.6: Hokuyo UTM-30LX lidar sensor in vertically nodding mechanism built at SSC Pacific (known as the Nodding Hokuyo) 16
Figure 3.7: 3D point cloud with radial ray in blue 16
Figure 3.8: 2.5D multi-level surface map grid representation (image from [Triebel 06]) 17
Figure 3.9: Representation of positive obstacle, step edge, and slope hazards 18
Figure 3.10: Lidar scan of a negative obstacle 20
Figure 3.11: Geometry of negative obstacle (ditch) detection 21
Figure 3.12: Negative Obstacle DetectoR (NODR) flowchart 22
Figure 3.13: Small ground vehicle overlooking step 23
Figure 3.14: A potential negative obstacle will have a gap with a distance greater than if the next laser point was found on a flat surface, and an increase in angle of γ times the angle difference. Points A and B are true returns from a lidar traced along a ray. The expected increase in vertical angle is Δ. If the surface was flat and the vertical angle was $\gamma * \Delta$, the next laser return would have been C. If the distance between A and B is greater than A and C, this could be a negative obstacle. 24
Figure 3.15: Ray tracing examples with results determined during the potential negative obstacle detection step 25
Figure 3.16: 3D point cloud returns from sequential frames. The area in the bottom middle of each image (indicated by a red box) has a steep negative slope. Figure 3.16a shows no detected signals from the slope and would be classified as a potential negative obstacle. In Figure 3.16b, more data exists and the area can now be correctly identified with the value of its slope. 27
Figure 3.17: Sensor angles used to determine the min and max distance from the vehicle a steep negative decline can be discovered. 29
Figure 3.18: Potential negative obstacle displays. This analysis uses the vertical lidar scan (ray tracing) to classify negative hazard areas. 30

Figure 3.19: Side view and top-down view of grid cells painted on the screen with various colors representing the classification features: brown is a horizontal cell (ground), blue is a potential negative obstacle, green is a step edge, yellow is a steep slope, and red is a positive obstacle. 32
Figure 3.20: Path planning arcs. The arc with the best traversability score will be chosen 33
Figure 3.21: The chosen arc (yellow) after traversability scores have been calculated 34

Figure 4.1: Simulated sample terrain (top) with lidar scans from both the small (middle) and large (bottom) 3D lidars 37
Figure 4.2: Negative obstacle rays detected correctly from NODR Only small lidar experiment 39
Figure 4.3: Negative obstacles detected correctly from NODR Only small lidar experiment 39
Figure 4.4: Detection images for the smooth and rough NODR and SVM experiments for the small lidar. The gaps in data are the negative obstacles, and most are colored by the detection method (NODR is blue and SVM is yellow). The colors are layered, blue is on top, yellow below that, and red (for ground truth) is under that so that some colors are not visible in this image. There are positive obstacles as well, and all the data behind it is occluded from the lidar. 41
Figure 4.5: Negative obstacles rays detected correctly from SVM and NODR small lidar experiment 41
Figure 4.6: Negative obstacles detected correctly from SVM and NODR small lidar experiment. The red ground truth line is hidden behind the gree svm line which is 100% accurate. 42
Figure 4.7: Negative obstacles rays detected correctly from NODR Only large lidar experiment 42
Figure 4.8: Negative obstacles detected correctly from NODR Only large lidar experiment 43
Figure 4.9: Detection images for the smooth and rough NODR and SVM experiments for the small lidar. The gaps in data are the negative obstacles, and most are colored by the detection method (NODR is blue and SVM is yellow). The colors are layered, blue is on top, yellow below that, and red (for ground truth) is under that so that some colors are not visible in this image. There are positive obstacles as well, and all the data behind it is occluded from the lidar. 44
Figure 4.10: False positives (yellow) reported for ranges that were beyond 16m for the simulated large lidar sensor 45

Figure 4.11: Negative obstacles rays detected correctly from SVM and NODR large lidar experiment 45
Figure 4.12: Negative obstacles detected correctly from SVM and NODR large lidar experiment. The red ground truth line is hidden behind the green svm line which is 100% accurate. 46
Figure 4.13: Google Maps satellite image of the off-road course with numbers ordered to show the route taken 46

LIST OF TABLES

Table 2.1: Robotic Vehicle Systems Comparison 10

Table 3.1: 3D Perception sensor comparison 13

Table 4.1: Maximum detection ranges of obstacles found in off-road test route 47

ACKNOWLEDGEMENTS

I am thankful for the support of Professor Mohan Trivedi and past and present students in the UCSD Computer Vision and Robotics Research laboratory.

I am grateful for the sponsorship of the Department of Defense SMART scholarship, for paying for my time, tuition, and books and making the dream of graduate school possible for a family man.

I would also like to thank Mike Bruch, Bart Everett, Ryan Halterman and so many others in the Robotics Branch at SSC Pacific for their support with ideas, design, and software.

And of course for my wife for being patient and supporting me through years of juggling school and work, and for always taking care of the details.

ABSTRACT OF THE THESIS

Off-road Obstacle Classification and Traversability Analysis in the Presence of Negative Obstacles

by

Jacoby Larson

Master of Science in Engineering Sciences (Electrical Engineering)

University of California, San Diego, 2011

Professor Mohan Trivedi, Chair

In order for an autonomous unmanned ground vehicle (UGV) to drive in off-road terrain at high speeds, it must analyze and understand its surrounding terrain in real-time; it must know where it intends to go, where are the obstacles, and many details of the topography of the terrain. Much research has been done in the way of obstacle avoidance, terrain classification, and path planning, Yet few UGV systems can effectively traverse off-road environments at high speeds autonomously. This paper presents algorithms that analyze off-road terrain using a point cloud produced by a 3D laser rangefinder, determine potential obstacles both above ground and those where the ground cover has a negative slope (negative obstacles), then plan safe routes around those obstacles. To classify negative obstacles, this research uses a combination of a geometry-based method called

the Negative Obstacle DetectoR (NODR) and a support vector machine (SVM) algorithm. The terrain is analyzed with respect to a large UGV with the sensor mounted up high as well as a small UGV with the sensor mounted low to the ground.

Chapter 1

Introduction

This chapter provides a brief overview of the research performed for this thesis and the contributions to the robotics community. An introduction to off-road autonomous driving is included.

1.1 Research Goals, Motivation, and Scope

A report by SSC Pacific [Holste 09] concerning the mobility of UGVs for dismounted marines provides a survey and analysis of current robotic technologies and concludes that there are significant weaknesses in each system, especially in the area of mobility in the face of hazardous terrain. This report also identifies that NATO's vital gaps are "moving in all terrain with tactical behavior in nearly all weather conditions" and "autonomous road following". A recent information paper written by the United States Marine Corps on April 2011 [USMC 11] states that the operational challenges articulated by deployed Marine Forces, with respect to UGVs, are "Robots are ill-suited to navigate the harsh terrain of off-road Afghanistan" and "that UGVs/robots might work in deliberate clearing operations in an urban environment, but the technology is not ready for operational use with dismounted patrols in rugged terrain."

There is an urgent need to enhance the mobility of dismountd Marines to increase speed, lethality, and survivability, all of which would require a large UGV. Furthermore, explosive ordnance disposal (EOD) technicians are still carefully maneuvering small UGVs by tele-operation using a "soda-straw" view (Figure 1.1)

Figure 1.1: "Soda-straw" video feed from a small ground robot

from the video feed displayed on a bulky hardened laptop (Figure 1.2). This technology lag is in large part due to lack of real-time autonomous off-road traversability analysis for unmanned ground vehicles (UGV), including negative obstacle (negative slopes, drops, or cliffs) detection at sufficient distances. Military UGV applications must accommodate off-road terrain based on the warfighting areas in which the US military is currently involved. The technology must also be able to handle the size of vehicles (large and small) that would address those needs. Accurately representing off-road terrain and analyzing it in real-time is a challenge for most UGV systems and the majority operate at slow speeds over relatively flat terrain.

There are significant improvements in autonomous obstacle detection and avoidance that need to be made before higher mission-oriented tasks can be accomplished in those areas of the world in which the US military is currently fighting, which is what this research addresses. This work can also be applicable to the area of urban road driving, providing obstacle detection, road analysis, and path following (Figure 1.3). However, the main thrust of this research is aimed at developing methods of safely traversing rough terrain at high speeds by: 1) detecting all of the obstacles that could be a serious threat, dependent on the size of the UGV, 2) producing traversability scores for each terrain location, and 3) planning a traversable route that avoids those obstacles and attempts to maintain a course towards a specified goal location.

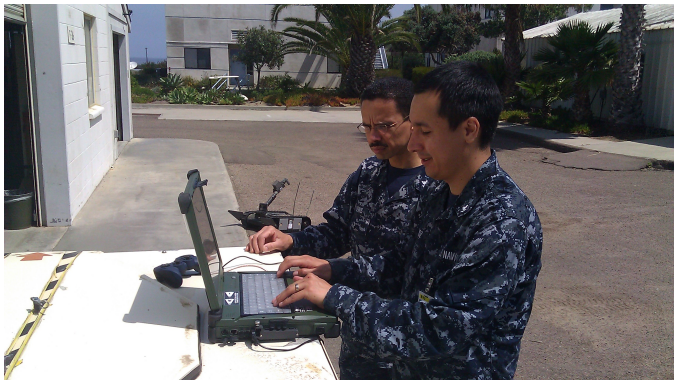


Figure 1.2: Navy reservists driving small UGV (in background)

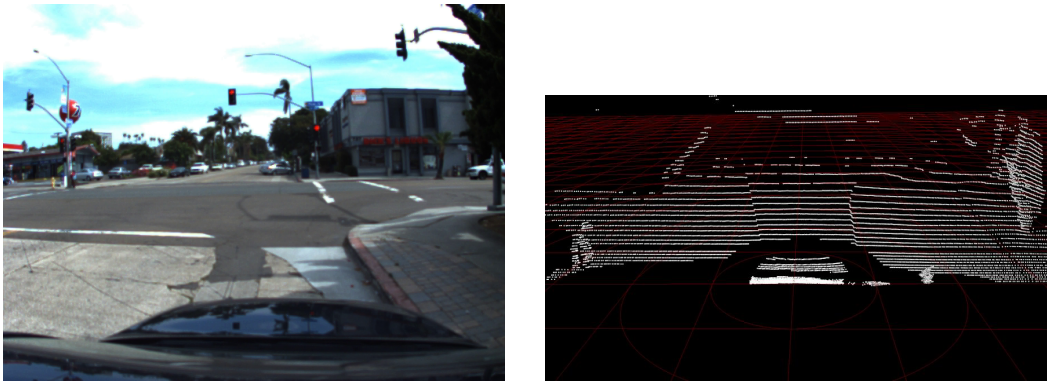


Figure 1.3: Urban traffic intersection: image (left) and point cloud (right)

1.2 Thesis Outline and Contributions

The Introduction of this thesis describes the reason this research has been performed. Along the way, many papers have provided support and inspriation to this work and their contributions are described in Chapter 2. The autonomous system created from this research is made up of a combination of technology previously described and some new work in the area of detecting negative obstacles. The approach used to analyze the 3-dimensional terrain and detect obstacles is detailed in Chapter 3. The new contributions of this thesis are a result of in-depth analysis of detecting and classifying negative obstacles from two lidar sensors, from the perspective of a large and small UGV. This paper details the hazards of negative obstacles, the complexities of detection based on geometry, and a step-by-step algorithm for detection called the Negative Obstacle DetectoR (NODR), as well as a machine learning method using support vector machines (SVM). The results of the NODR and SVM methods can be found in Chapter 4 for both a simulated environment and a real off-road course. Most research in this area has very few detection results, mainly because it is usually performed in a real off-road environment, which is almost impossible to precisely quantify or obtain ground truth. A simulated environment has been created to quantify the detection approach. Chapter 5 summarizes this research and provides ideas for future work.

Chapter 2

Related Research

Obstacle detection and avoidance for UGVs is essential for robotic vehicle mobility and as such, it has been researched and demonstrated with numerous sensors on a multitude of platforms for various terrains.

2.1 Video-based vs Lidar-based

Two of the main sensors used for gathering 3D information for obstacle detection for unmanned ground vehicles are stereo cameras and lidar. Both sensors have their advantages and disadvantages, and usually a combination of the two make up a successful solution. The advantages of stereo cameras are that they have low cost, do not emit electromagnetic signatures, and can incorporate color information easily. Color images facilitate classification of a few distinct classes such as grass, vegetation, soil, rock, and bark [Manduchi 04]. On the other hand, stereo cameras are computationally complex at higher resolutions, require careful calibration [Belongie 04], bound to a rigid camera-to-camera configuration, are sensitive to environmental conditions (especially outdoors) such as light saturation and color constancy [Barnard 02], require a larger baseline for increased range, have a non-linear range output, and have a limited view. Lidar systems can provide up to 360 degrees of viewing angle without losing resolution, produce high quality range data, and require little computational time and hardware to return a point cloud. Some of the disadvantages of lidar include the lack of color in the point cloud returns and the cost. Many of the unmanned systems that competed in the

DARPA Grand Challenge and DARPA Urban Challenge used a combination of lidar and cameras for obstacle detection and avoidance.

2.2 Military vs Consumer Systems

A large push has been made of recent years on the military side for unmanned vehicles. In fact, in the 2001 Defense Authorization Act, congress set a goal to have one third of the operational ground combat vehicles be unmanned by 2015 [Congress 01]. The main players for unmanned ground vehicles have been military partnerships with universities and a few companies, such as the DEMO I, II, and III projects [Shoemaker 98] and the DARPA Grand Challenge and Urban Challenge to name a few. SPAWAR Systems Center (SSC) Pacific has also been a major player in robotics since the early 1980's developing air, ground, water surface, and underwater unmanned vehicles, researching areas such as telepresence, multi-robot collaboration, human-machine interaction, computer vision, obstacle detection and avoidance, SLAM, human detection, stair climbing, and much more. A few commercial car companies have started integrating autonomous technologies into their vehicles, such as automatic parallel parking and collision warnings, for public use. But only recently have private companies shown interest in creating their own autonomous vehicles without the military's involvement, as for example Google's announcement in 2010 [Thrun].

2.3 Off-road vs On-road

There are major differences in obstacle detection and avoidance from unmanned vehicles in off-road environments versus those operating in urban environments. The obstacles in an urban setting are going to have flat surfaces like buildings, and almost everything that sticks up out of the ground should be considered an obstacle. Off-road, obstacles rarely have flat surfaces, and vegetation might or might not be a true impediment to the vehicle, depending on the vehicle's size. Another difference is the traversable paths. In the city, there are only a certain number of routes that can be chosen, and almost all of them assume the

vehicle will stay on the road. Even the 2005 DARPA Grand Challenge, an off-road autonomous vehicle competition, which had sharp switchbacks in the Beer Bottle Pass, was a course made up of roads. Standford's Stanley, the vehicle that won the 2005 competition, admits it was built "...on a long history of research of road finding" [Thrun 06]. Such urban systems usually don't need to worry about the traversability of the terrain, the slope, or negative obstacles that are found in off-road scenarios.

2.4 Mapping

For off-road terrain, a simple 2D obstacle map most likely will not provide enough information for traversability planning, yet storing accurate 3D models takes up too much precious space in memory on typical embedded computers found on robots. Moreover, a priori environment data can be very useful, but not complete for a robot in off-road terrain where new vegetation growth, a freshly dug hole, or a recently placed pipe could cause an accident and leave the robot in pieces. To reliably detect navigation hazards (positive obstacles, negative obstacles, steep slopes, and step edges), understanding the environment should be done, at least in part, by on-board sensors. It is important to consider how to store that data in a way that is accurate enough for navigation, yet fast enough to search and analyze in real-time.

The point-cloud data retrieved from the sensor systems can be modeled using a 2D, 2.5D [Fong 03], or 3D map. A 2D grid map, also known as an occupancy grid, is usually used to report the probability of finding an obstacle in the grid cell. These systems are of great use for path planning and navigation since a robot is very interested in the location of obstacles and the representation is quick to search. The 3D grid map is made up of voxels that can be very useful for path planning in air and under-water applications [Carsten 06], but can take up a considerably large amount of memory. Because of the complexity and time required to search through a 3D-voxel grid map, a very common method of extraction is to model the terrain in a 2D grid with elevation information, referred to as a 2.5D map. Once information is obtained and stored in a map, there is a need to analyze that map

and classify those areas that are the most hazardous.

2.5 Classification

Classification of the terrain allows the vehicle to determine the obstacles in the environment and facilitates a path planner to find a safe path to a goal location. Terrain classification can be done in image space as in [Rankin 09], or in map space. It makes more sense to classify the terrain in map space for a 3D lidar point cloud, after it has been organized into a 2.5D map. Those hazards that are of the most concern for UGVs are negative and positive obstacles.

Negative obstacles are difficult to detect, especially at long ranges, but methods used have included searching for negative slopes that are too steep, or gaps in data that exceed a distance threshold followed by a drop in elevation or a steep uphill slope [Seraji 03, Murarka 08]. Rankin et al. use both a column detector for gaps that exceed a width and height threshold and then follows up with a region size filter to eliminate negative obstacles that are too small as well as a unidirectional elevation difference detector. Aerial images and lidar data have been demonstrated [Silver 06] to do negative obstacle detection as well, which can detect the bottom of the negative obstacle, which is not always the case from the perspective of a ground robot. In [Hong 98, Heckman 07], ray tracing is perfomed from the current position of the laser, and context-based labeling from occlusions (from the ground or positive obstacles) are considered while detecting negative obstacles. JPL also has presented a novel method for detecting negative obstacles using thermal signature for night-time detection [Matthies 03].

However, not all objects that take up space in a 3D point cloud are obstacles that would stop a vehicle, typical examples include weeds, tall grass, branches, and other vegetation. Carnegie Mellon University has demonstrated a method using 3D lidar to classify natural terrain into saliency features such as scatter, linear, or surface, which can be used for further traversability analysis [Lalonde 06]. JPL also reported being able to detect large obstacles such as rocks behind tall grass using a maximum-value range filter [Manduchi 04].

One of the methods that will be used in this work to classify those obstacles

is a support vector machine, a binary non-probabilistic linear classifier that can be used for a two-group classification problem, and in this case, to classify negative obstacles. This is done by creating a separating hyperplane that maximizes the minimum distance to the training data samples [Vapnik 95], and then provides a prediction of the location of the next sample with respect to that hyperplane, based on its input parameters.

2.6 Traversability

Each grid cell, once classified, can receive a cost value for traversability [Rankin 09], which will help for path planning and navigation. This cost can be a binary "go" or "no-go" value, but that doesn't allow advanced path planning methods to utilize the full potential of the vehicle model, such as when there is a ditch in the road, which would be classified as "no-go", but in actuality the baseline of the separation between vehicle wheels is wide enough to straddle it. The Morphin algorithm [Simmons 95] analyzes the terrain using three measures: slope, roughness and step height. The slope can be calculated with a least squares plane fitting, and the roughness is the residual error from the plane fit. The traversability cost is determined by the worst of the three values.

The method of approach in this research paper combines many of these previous works, and builds upon them in an attempt to create a more accurate obstacle detector at further ranges to allow a vehicle to travel at higher speeds.

2.7 Robotic System Comparison

A comparison of multiple unmanned vehicle systems is shown in Table 2.1.

Table 2.1: Robotic Vehicle Systems Comparison

Vehicle	Obs (-)	Obs (+)	>32kph	Off-road	# Perception Sensors
This research	x	x	x	x	1
Stanford Stanley		x	x		8
NREC Crusher		x	x	x	29
JPL Mars Rovers		x		x	2
JPL Demo III	x	x		x	6
CMU Sandstorm		x	x	x	10
CMU Boss		x	x		18
TORC GUSS		x		x	3
TORC ByWire XGV		x	x		4
iRobot R-Gator	x	x	x	x	6
Google Car		x	x		1

Chapter 3

Approach

This section describes the method to achieve autonomy, Figure 3.1. This method depicts the steps necessary to for the robot to perceive, analyze, and act, they include:

- Data capture
- Pre-processing
- Classification (obstacle detector)
- Traversability analysis
- Path planning
- The robot drives

For this method to be useful for a fast moving vehicle, it must be completed quickly (under 1s). For reference, a vehicle traveling 32kph will move 9m in 1s. To understand how fast an autonomous off-road vehicle can really go, one must also consider how long it takes for the vehicle to stop.

The stopping distance for a vehicle traveling at a certain velocity v can be determined using Equation 3.1, referenced in [Matthies 94] and [Matthies 03].

$$R = \frac{v^2}{2\mu g} + vT_r + B \tag{3.1}$$

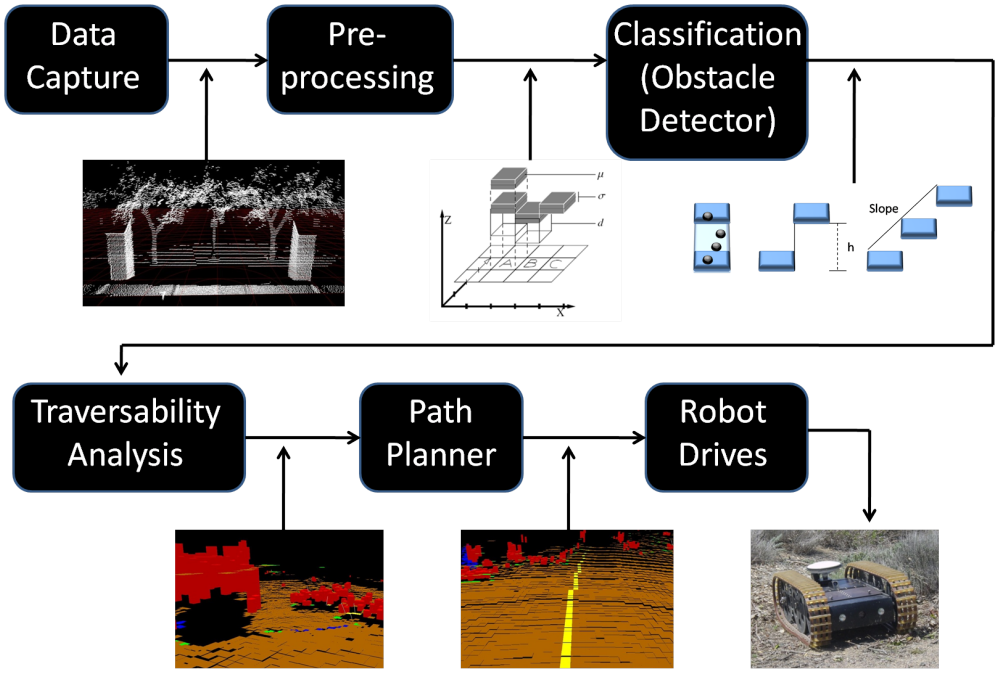


Figure 3.1: Method for Autonomy

where μ is the coefficient of static friction between ground and wheels with a common value of 0.65 for off-road driving, g is gravitational acceleration with a value of 9.8m/s^2, T_r is the total reaction time with a common value of 0.25s, and B is a buffer distance used for safety with a value of 2m in these experiments. The velocity value becomes the dominant term at $v > 3.2$m/s: for a velocity of 24kph (15mph), the distance needed to stop is 7.2m; for a velocity of 48kph (30mph), the distance needed to stop is 19.4m (see Figure 3.2).

3.1 Data Capture

The method for autonomy starts with data capture of the surrounding environment by a perception sensor. For off-road terrain, is is important that this sensor perceive the surrounding environment in all three dimensions, to provide an understanding of the pitch and roll of the terrain. The requirements of this research led to the selection between 3D lidar and stereo cameras. Table 3.1 lays out the advantages and disadvantages of both classes of sensors.

For this research, a 3D lidar sensor was used for the data capture portion of

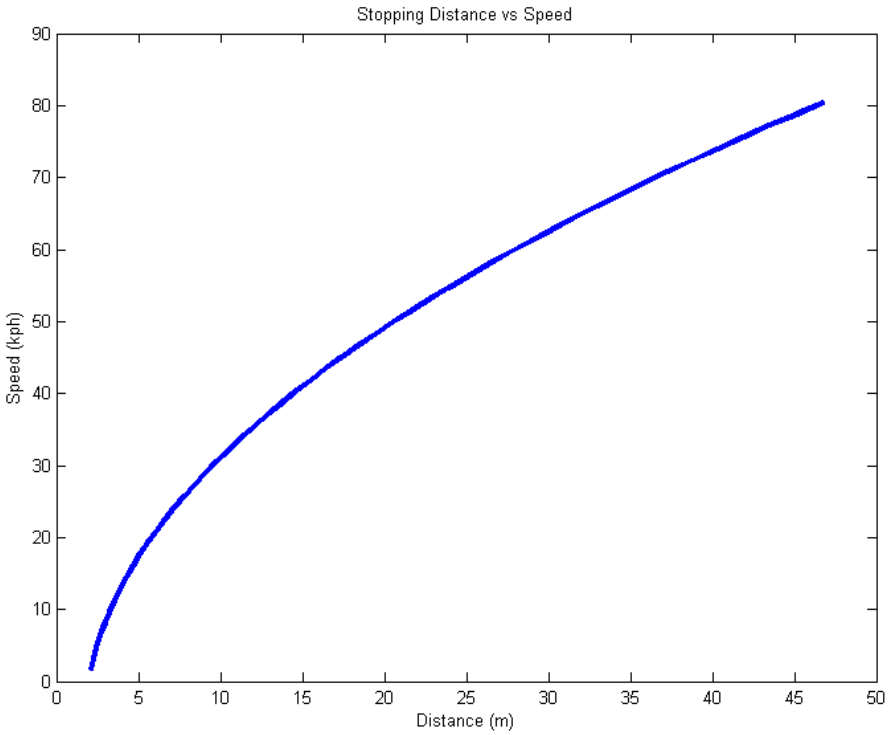


Figure 3.2: Stopping Distance vs Speed

	lidar	stereo
Pros	work at night	inherent color
	360° view	cheap
	less processing	no electromagnetic signatures
	linear resolution	angular resolution
Cons	expensive	more processing
	slow 3D rotation	light sensitive
	no color	calibration
	few libraries	limited view

Table 3.1: 3D Perception sensor comparison

the method for autonomy. It will allow easy data capture at night, view a wide azimuth range, and provide a 3D point cloud without significant processing or additional computers.

This research was conducted to fit both a large UGV that can travel at high speeds, greater than 32kph, as well as a small UGV platform that travels at 2.5m/s. The sensor used for the large platform required a greater detection distance and the sensor for the smaller platform required a small form factor. The intended large platform is a Max ATV (Figure 3.3), a six-wheel skid-steer all-terrain vehicle with dimensions of length 2.6m, width 1.5m, and height 1.7m, including a roll bar. It weighs 805lbs and can carry up to 1605lbs and tow 1000lbs. The 3D lidar used on this large UGV platform is a Velodyne HDL-64E (Figure 3.4). This lidar system provides readings of range and intensity out to a distance of 120m with 80% reflectivity, providing 100,000 data points with 360° horizontal and 26.8° vertical field of view at a rate of 10Hz. SSC Pacific has demonstrated the accuracy of this sensor on the water surface by detecting a lobster trap out to 40m and a partially submerged black rock out 65m [Halterman 10]. The smaller ground vehicle platform is an iRobot Packbot (Figure 3.5) with length 89cm, width 52cm, and height 18cm, mounted with a Hokuyo UTM-30LX lidar sensor, set in an after-market mechanism that nods it vertically for a full 3D scan, referred to as the Nodding Hokuyo (Figure 3.6). The Nodding Hokuyo scans 270° horizontally and can rotate slow enough to have a 2.5° vertical angular resolution, with a pitch from -90 degrees to +90 degrees.

3.2 Pre-processing (Representation of the Point Cloud)

The data is pre-processed to facilitate and speed up the classification process and has two different representations. The first representation is a set of vertically aligned laser rays that are used for negative obstacle detection as seen in Figure 3.7. This uses the radial structure of the two lidar sensors and stacks each horizontal scan line on top of each other. The ray can be traced from the sensor or a point in the first horizontal scan out toward a point in the last horizontal scan and makes

Figure 3.3: Max ATV UGV test-bed platform

Figure 3.4: Velodyne HDL-64E lidar sensor

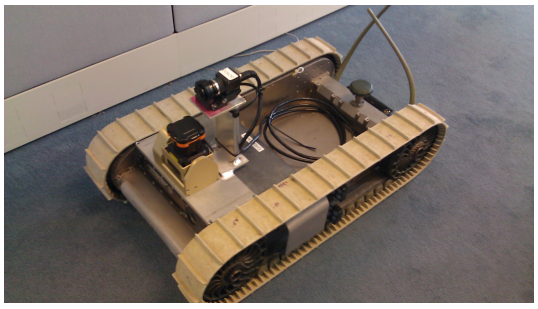


Figure 3.5: iRobot Packbot UGV test-bed platform with mounted Nodding Hokuyo sensor and color camera

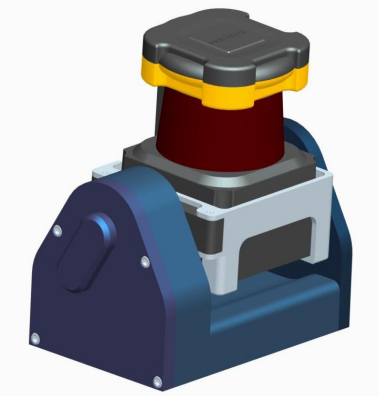


Figure 3.6: Hokuyo UTM-30LX lidar sensor in vertically nodding mechanism built at SSC Pacific (known as the Nodding Hokuyo)

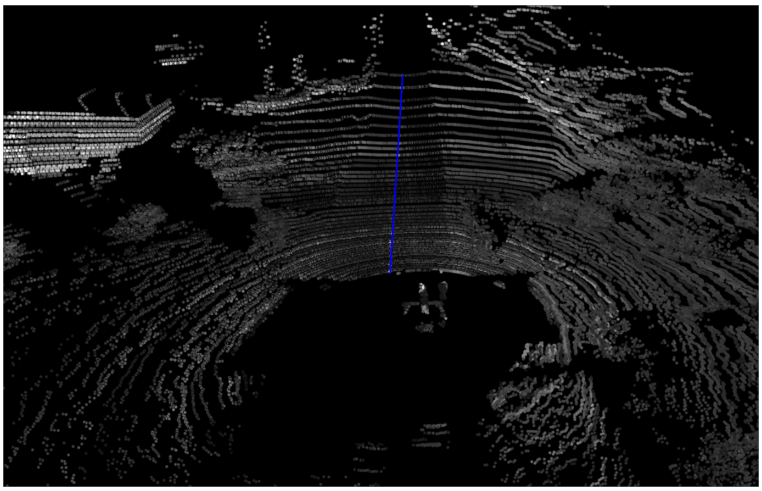


Figure 3.7: 3D point cloud with radial ray in blue

it easier to calculate slope and find discontinuities. The second representation is a 2.5D multi-level surface map in Figure 3.8. When the elevation coordinate is ignored, the 3D laser points can be arranged into a 2D Cartesian grid of 40cm x 40cm cells for the large UGV, measuring 100m in the x and y directions, and 12cm x 12cm cells for the small UGV, measuring out 30m in the x and y directions. For reference, this paper uses a right-handed coordinate system where x is lateral, y is longitudinal, and z is vertical through the ground plane. Each cell in the grid contains additional information, such as elevation, terrain traversability measures, and so on. Although a simple 2.5D grid can be very efficient for determining ground cover, it is limited in its ability to detect overhead obstacles like overhanging trees or pathways such as bridges. The multi-level surface handles these situations.

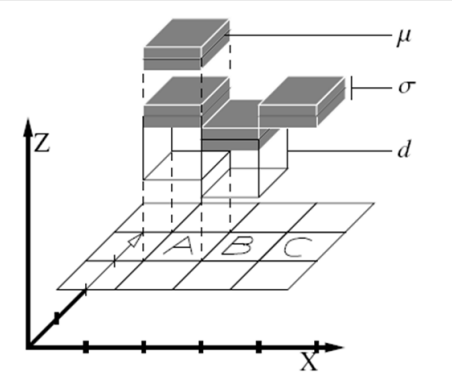
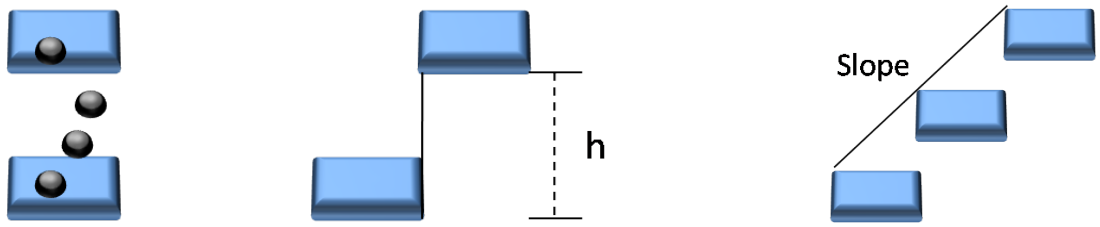


Figure 3.8: 2.5D multi-level surface map grid representation (image from [Triebel 06])

A multi-level surface map, as discussed in [Triebel 06], allows multiple surface levels to be represented in one grid cell, which can aid in correctly classifying overhanging obstacles and provide additional search paths if one of those surfaces is a bridged road above the robot. Each surface retains statistical information of the points that lie in it, such as elevation mean and variance, as well as the number of points that fall within the cell, with the maximum and minimum elevation. Those surface levels where the 3D points span some vertical area, large enough to be classified as a positive obstacle, are called a vertical surface. Otherwise, the surface is referred to as a horizontal surface. If a point is greater than a height threshold (usually the height of the robot) above a surface level of a grid cell, then it would be placed in its own surface level. Because of the mutliple surface levels, the traversability component of the robot may plan routes through grid cells with multiple layers. These surface levels are ordered from lowest to highest in the cell for fast lookup when adding new data points. If points are added that lie between two surface levels and now the difference in elevation between the two surfaces is smaller than the height of the robot (meaning the robot will hit something), then those levels are merged into one and variances, means, and depth values are recalculated.

The elevation of the data points that fall into the surface level are represented as a Gaussian distribution. For fast processing and to reduce the amount of memory used, online calculations of mean and variance are used. This work used a method for calculating an unbiased estimate of the population variance (Equation 3.2).



(a) Positive obstacle representation

(b) Step edge obstacle representation

(c) Slope representation

Figure 3.9: Representation of positive obstacle, step edge, and slope hazards

$$\sigma^2 = \frac{\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2/n}{n-1} \tag{3.2}$$

The variance can easily be computed and merged with other surface levels by keeping track of the number of points that fall into this surface level, the cumulative sum of elevations, and the cumulative sum of the squares of the elevations.

A positive obstacle is classified when the elevation variance of a surface level (calculated when the 3D data points fall into the surface level of a cell as in Figure 3.9a) exceeds a variance threshold: this method ignores extreme outliers with a large enough sample size. This variance threshold is based on the square of the standard deviation, which is half the the maximum height the wheels can traverse (also known as the step edge threshold).

Another hazard feature called a "step edge" can be detected by calculating the elevation differences between cells, such as in Figure 3.9b, which is UGV-size dependent. If the UGV is small with small tires or treads, then even a small gap in elevation could block or tip over the robot.

Additionally, a steep slope can be an obstacle to a robotic vehicle. The slope of each cell is calculated by fitting the points of the surrounding cells to a plane (Figure 3.9c) and finding the surface normal of that plane, which can be found using principal component analysis (PCA). This analysis uses an orthogonal transformation to represent the data in terms of the variance in each of the three dimensions: x, y, and z. PCA can be computed either by eigenvalue decomposition of the covariance matrix or the singular value decomposition (SVD) of a data matrix. This data matrix is usually mean-centered, $(x_1 - \bar{x})$, to find the component

of greatest variance and is shown in Equation 3.3. After SVD, the surface normal of the plane should be the left singular vector that corresponds to the smallest singular value. At this point in the analysis, all the of individual points in the cloud point have been erased and only the statistical information for each surface level of the cells remain. The data matrix can be calculated using the mean of the elevation (the z component) and the x and y values of each of the grid cells. Note that it is not required to calculate the slope of the cells already classified as obstacles, since the vehicle will not be traveling over them. This method uses the eight connected neighbors of each cell, and only calculates the slope if there are at least four neighbors. The $\bar{x}$, $\bar{y}$, and $\bar{z}$ represent the mean values of the x, y, and z values of the cell's neighbors.

$$M = \begin{bmatrix} (x_1 - \bar{x}) & (y_1 - \bar{y}) & (z_1 - \bar{z}) \\ (x_2 - \bar{x}) & (y_2 - \bar{y}) & (z_2 - \bar{z}) \\ \cdots & \cdots & \cdots \\ (x_n - \bar{x}) & (y_n - \bar{y}) & (z_n - \bar{z}) \end{bmatrix} \tag{3.3}$$

If the absolute value of the difference between the surface normal and the gravity vector is too large, it is classified as a slope obstacle.

After the terrain analysis, each cell is represented as empty or unkown, a horizontal cell (usually the ground plane), a positive obstacle, a step edge, a steep slope, a potential negative obstacle, and/or a true negative obstacle. A cell may have more than one classification, but those that are the most hazardous take priority when calculating terrain traversability. The negative obstacle classification step is performed before any of the other steps and is described more thoroughly in Section 3.3.

3.3 Negative Obstacles

Negative obstacles are ditches, cliffs, drop-offs, holes, or terrain with a steep negative slope that if traversed would cause damage to the vehicle. Negative obstacles can be just as hazardous to unmanned vehicles as obstacles above ground because they could cause roll-over, tip-over, or high-centering, but are usually much

Figure 3.10: Lidar scan of a negative obstacle

harder to detect. In most cases they show up as an absence of data, as in the 3D lidar scan seen in Figure 3.10. Usually ditches that are equal to or greater than the width of the diameter of the wheel are enough to cause damage to a vehicle. Negative obstacles of greater widths may sometimes be crossed by vehicles at high enough speeds, but this paper will not attempt to provide velocity modification techniques, simply methods of detection.

3.3.1 Detection Range

Some negative obstacles are difficult to detect when close up and nearly impossible from far away. Equation 3.4, based upon the small angle approximation and referenced from [Matthies 03], illustrates the difficulty of detecting negative obstacles at a range R and is shown in Figure 3.11. The width of the obstacle is w, H is the height of the sensor from the ground, h is the depth of the obstacle seen by the sensor, and R is the range from the sensor to the obstacle. The equation to solve for θ is

$$\theta \approx \frac{Hw}{R(R+w)} \tag{3.4}$$

The angle θ decreases significantly as the range increases ($\sim\frac{1}{R^2}$), which makes

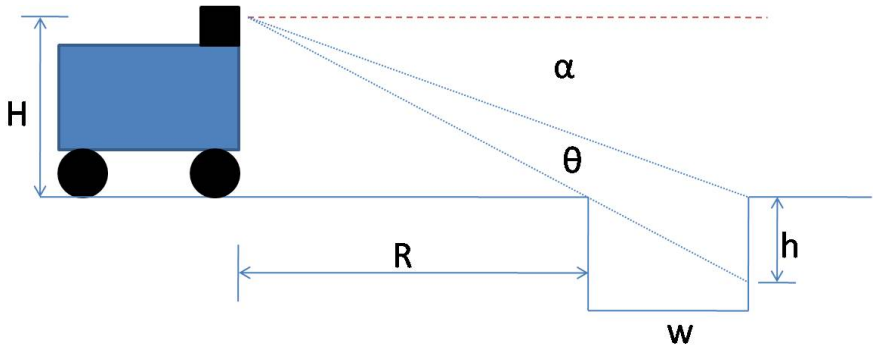


Figure 3.11: Geometry of negative obstacle (ditch) detection

negative obstacles so difficult to detect with increasing range. Yet detecting negative obstacles at greater distances is essential, especially for fast moving UGVs. Two different methods to detect negative obstacles were used. The first method uses a support vector machine (SVM) that must be trained with ground-truth data. It has been expected that there would be a limit in the range of correct classification for the SVM because of the parameters passed into it. The second method is called the Negative Obstacle DetectoR (NODR), which uses a number of filters and looks for contextual cues, so it can have expanded range benefits.

3.3.2 NODR Classification Approach

Because of the difficulty in detecting negative obstacles, this classification method errs on the side of generously detecting negative obstacles and then labeling them as only potential negative obstacles. Only when the potential obstacle comes within close range to the vehicle can enough data be gathered to truly classify it as a real negative obstacle. This geometry-based method for detecting negative obstacles is called Negative Obstacle DetectoR (NODR) and is shown in Figure 3.12. The range for true classification is described later on in Section 3.3.4.

The NODR classifies potential negative obstacles by detecting steps or gaps, an absence of data, where there could exist a ditch, cliff, or negative slope. The detection starts by tracing a ray of 3D points outward from the sensor, following the returns from the vertical alignment of lasers starting with the lowest vertical angle towards the highest angle. This algorithm is based on laser sensors that produce structured results, such that can be vertically alligned. The Hokuyo UTM-30LX

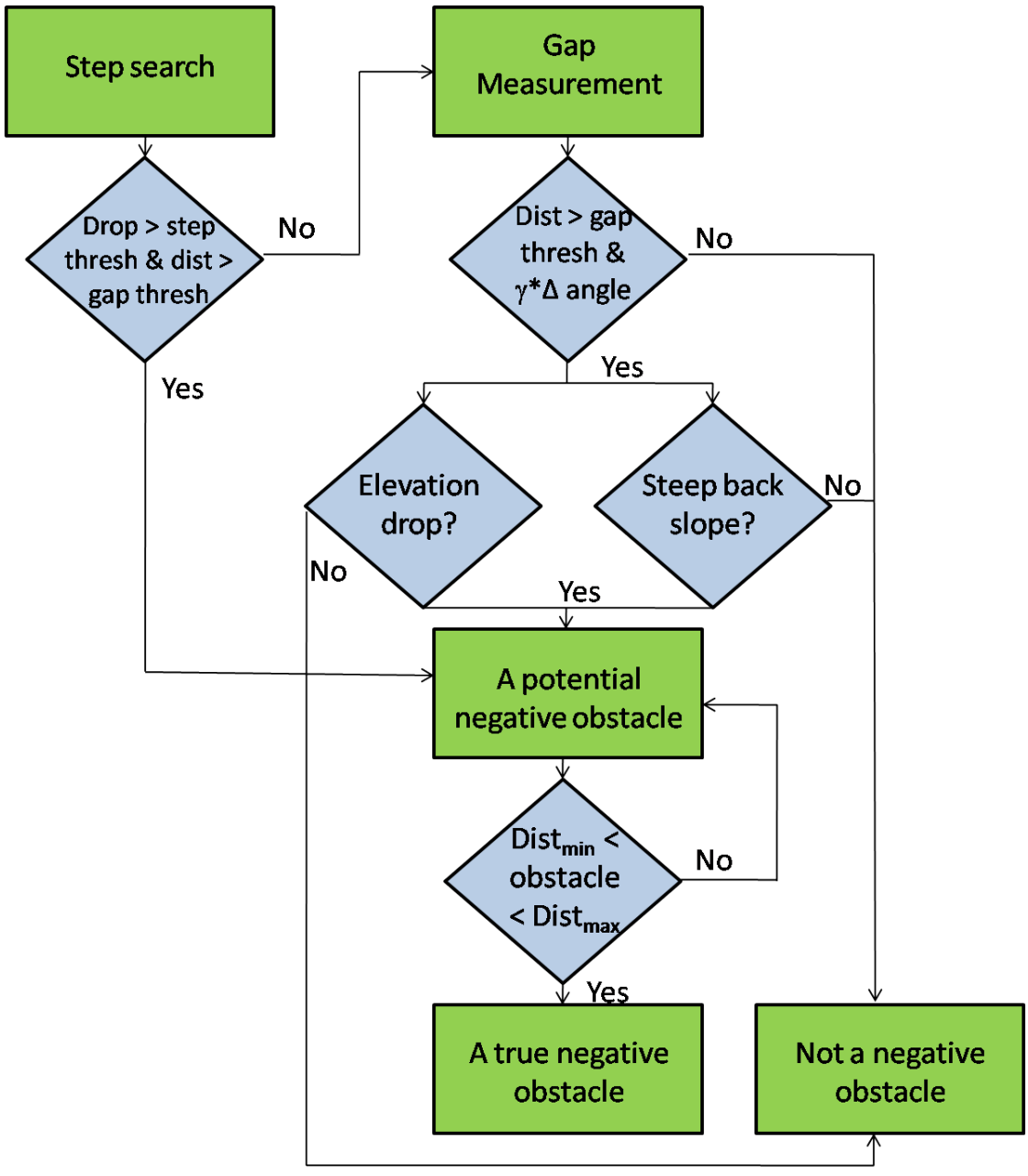


Figure 3.12: Negative Obstacle DetectoR (NODR) flowchart

Figure 3.13: Small ground vehicle overlooking step

returns a maximum of 1080 lidar beams per horizontal scan and the vertically nodding mechanism allows the Hokuyo to produce another horizontal scan above or below the previous scan. This aligns well for following a radial ray of lidar beams. The lidar beams from the Velodyne HDL-64E, on the other hand, need to be modified slightly. This sensor has 64 vertically aligned lasers but reports 3 laser pulses in the top 32 lasers for every 1 pulse in the bottom 32 lasers. To handle the ray tracing in this case, the 3 laser returns from the top 32 lasers have been combined into one return by taking the average x, y, and z values of the top 32 lasers and combining them with the bottom 32 lasers, providing one complete radial ray of 64 lasers (Figure 3.7).

The first step used in the NODR method is to search for a step, a drop in elevation beyond a step threshold, that extends beyod the gap distance, as in Figure 3.13. The gap threshold of the two UGVs are determined by the size of a hole that would cause damage or stop the platform. For those vehicles with wheels, it is usually the wheel diameter. For the Max ATV with three wheels on each side, assuming the center of gravity is over the center wheel, it is the distance from the front wheel to the center of the center wheel, which is 1.1 meters. For tracked vehicles like the iRobot Packbot, it is the distance from the front of the treads to the center of gravity, at which point the vehicle begins to fall forward. The Packbot can stretch its reach by extending its front flippers, making the gap threshold to be about 54 cm. Note: no center of gravity measurments were taken for either of these two ground vehicles and these gap threshold values are only estimates to be used in siimulated environments.

If the step search does not return a negative obstacle, then the NODR looks for gaps in data between two points. While tracing these radial rays, if a gap is found with a distance greater than a gap threshold then it looks at the gap angle.

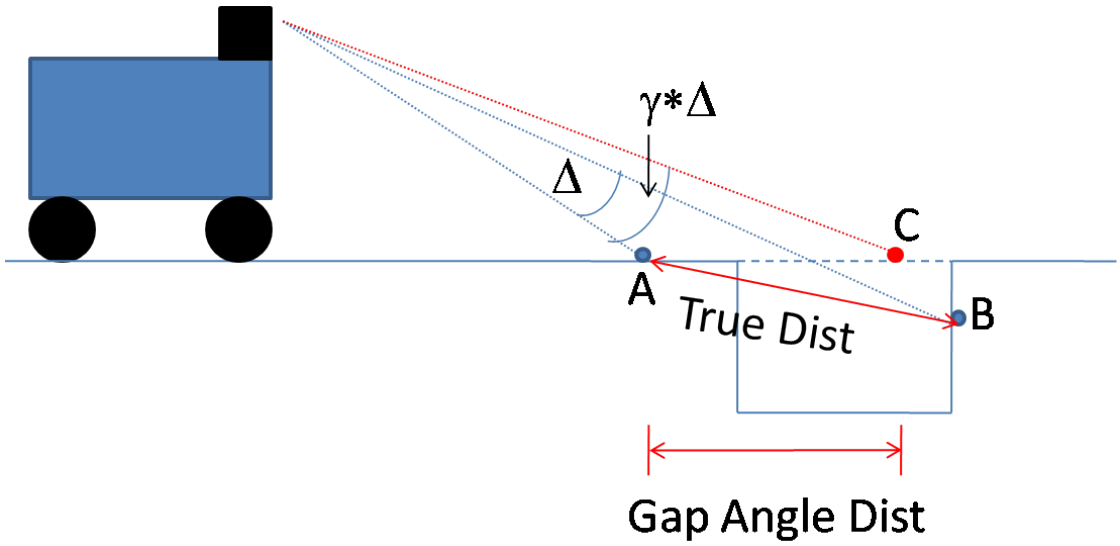


Figure 3.14: A potential negative obstacle will have a gap with a distance greater than if the next laser point was found on a flat surface, and an increase in angle of γ times the angle difference. Points A and B are true returns from a lidar traced along a ray. The expected increase in vertical angle is Δ. If the surface was flat and the vertical angle was $\gamma * \Delta$, the next laser return would have been C. If the distance between A and B is greater than A and C, this could be a negative obstacle.

The gap angle filter uses the increase in vertical angular resolution between the two scans. The threshold here is the distance that would be expected if the next point had the same elevation as the previous point, and the vertical angle had increased by $\gamma * \Delta$ where Δ is the vertical angular resolution between horizontal scans (Figure 3.14). Tests provided the best results when γ had a value of 1.5. It is noted that this might not be the most accurate method to calculate true negative obstacles, but this algorithm is attempting to be conservative to avoid obstacles, and as was emphasized from Equation 3.4, negative obstacles are extremely difficult to detect at long ranges. Furthermore, if a gap is sufficiently wide to pass these gap filters, the algorithm looks for contextual cues.

The cues to look for are either a significant drop in elevation, such as one that would appear if the slope was steeper than the maximum declining slope threshold (Figures 3.15a and 3.15b), or the data after the gap has a significant positive slope (Figure 3.15c), as in the sloping up-side (the back) of a ditch. The slope of the up-side of a ditch can be determined by first calculating the number of following points should be part of the back slope, determining their slope, and threshold

(a) Gap followed by a large drop in elevation: potential negative obstacle

(b) Slight uphill slope followed by a gap and drop in elevation: potential negative obstacle

(c) Gap followed by a steep uphill slope: potential negative obstacle

Figure 3.15: Ray tracing examples with results determined during the potential negative obstacle detection step

that value. The number of points is calculated by finding the viewing angle of the negative obstacle, which for Figure 3.11 is θ. For ranges R that are not quite so distant, where the small angle approximation doesn't work and equation 3.4 doesn't provide an accurate measurement of θ, this equation is

$$\theta = tan^{-1}(\frac{H}{R}) - tan^{-1}(\frac{H}{R+w}) \tag{3.5}$$

The number of slope points to be used is $num\_points = \frac{\theta}{\Delta}$, where Δ is the vertical angular resolution of the sensor. As long as num_points is greater than 1, the the back slope can be determined. When the gap is detected at ranges where a back slope is impossible to calculate because the vertical angle resolution is less than θ, it simply ignores this filter.

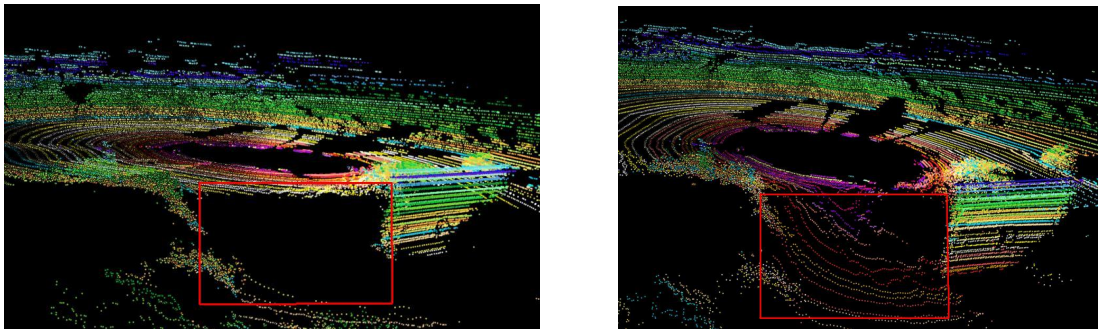
The other instances where it is important to detect negative obstacles are when the gap is detected at the beginning of the ray or at the end of the ray. Those

negative obstacles found at the beginning of the ray, meaning a large gap is found even before the first measurment, are very difficult to analyze, since there really is no data for that gap. It might be a negative obstacle or it might simply be a downward slope that can be traversed by the vehicle but the sensor cannot tilt enough to detect it. More about this is addressed in Section 3.3.4.

When there are multiple lidar scans in the end of the vertical ray trace that are missing, it is possible that there is a large negative obstacle whose end state is not detectable by this sensor (because the gap goes beyond the maximum range). There can be a threshold for the number of scans missing and the distance from the sensor of the last reading that will classify the rest of the ray as a negative obstacle, all the way out to the maximum range of the sensor.

The number of classified negative obstacles could be reduced by not classifying those gaps that follow a positive obstacle, but this is not always beneficial. This research concluded that it does not add value to differentiate between the gaps caused by a true negative obstacle or just occlusion from a positive obstacle. A vehicle would not navigate through a positive obstacle anyway, and processing the potential negative obstacles is trivial, whereas detecting positive obstacles and removing potential negative obstacles found afterwards can take away many clock cycles of precious processing time.

The result from the NODR is a vector ray between two laser points that can be used to populate cells in the 2.5D grid map, marked as potential negative obstacles and used to plan paths to avoid obstacles. The approach that has been taken in this research is not to completely avoid these detected "potential" obstacles. They might only be steep slopes that are traversable, but these sensors do not have complete information because of the remoteness of the vehicle and limitations of the geometry-based methods. In most cases the vehicle must be much closer to actually classify these as true negative obstacles. The proposed method is to slow down when these potential negative obstacles are in the immediate path, and avoid them once they have been truly classified.



(a) (b)

Figure 3.16: 3D point cloud returns from sequential frames. The area in the bottom middle of each image (indicated by a red box) has a steep negative slope. Figure 3.16a shows no detected signals from the slope and would be classified as a potential negative obstacle. In Figure 3.16b, more data exists and the area can now be correctly identified with the value of its slope.

3.3.3 SVM Classification Approach

As with the NODR classification approach, the support vector machine returns the vector rays in between laser points that are classified as potential negative obstacles. Parameters passed into the SVM include the range to the first point (of a pair of two radially aligned points) from the sensor, the distance between the two points, the change in vertical angle of the two points (as if the second point had the same elevation value as the first point) in reference to the vertical angular resolution (Figure 3.14), and the elevation difference of the two points. The SVM is trained by many samples of ground truth vector rays of negative obstacles as well as non-negative obstacles, then tested on each pair of vertically aligned points from the sample data.

3.3.4 Real Negative Obstacle Classification

As an example, Figure 3.16a depicts a large gap in data that would be tagged as a potential negative obstacle. However, when that area is within range of real negative obstacle classification (Figure 3.16b), the increased number of data points reveal the area is a steep downhill slope that is within the threshold of slope traversability.

Those potential negative obstacles that are within a narrow range of values are

elevated to a true negative obstacle condition and given a prohibitive traversability score. This range is calculated based on the maximum negative slope the vehicle can traverse and the vertical angle measurements from the sensor, found in Equation 3.6 and in Figure 3.17. For instance, the maximum negative slope allowed in these tests for both of the robotic vehicles, ϕ, is a 20° decline. Based on the maximum negative vertical angle θ_{max} of the sensor and the angle Δ between each increasing horizontal scan, the sensor would only be able to detect a steeper decline between $Dist_{min}$ and $Dist_{max}$. For the large sensor on the large UGV platform, this range is between 4.76 meters and 5.91 meters. After $Dist_{max}$, the angle of the lidar scan with respect to the horizontal is less than ϕ and would not actually detect the slope. Therefore, those potential negative obstacle rays that intersect with the range $Dist_{min}$ and $Dist_{max}$ are considered real negative obstacles. This is a very short distance to react to a real negative obstacle, but the path planning module should have already slowed down as it was approaching the potential negative obstacles (the speed should be slow enough to allow stopping distance before the obstacle, using Equation 3.1). It should also be pointed out that because of the lack of data between the vehicle and $Dist_{min}$, it is very difficult to determine if a gap found before the first measurement is because of a negative obstacle or a traversable declining slope. Therefore, these gaps would not be classified as real threats, and because of this limitation in close range negative obstacle detection, it is important to either tilt the sensor on the x axis to detect negative obstacles extremely close to the vehicle or keep a map history of previous obstacles and current localization in that map.

$$\begin{aligned} Dist_{min} &= \frac{H}{tan(\theta_{max})} \\ Dist_{max} &= \frac{H}{tan(\theta_{min})} \\ \theta_{max}, \theta_{min} &> \phi \end{aligned} \tag{3.6}$$

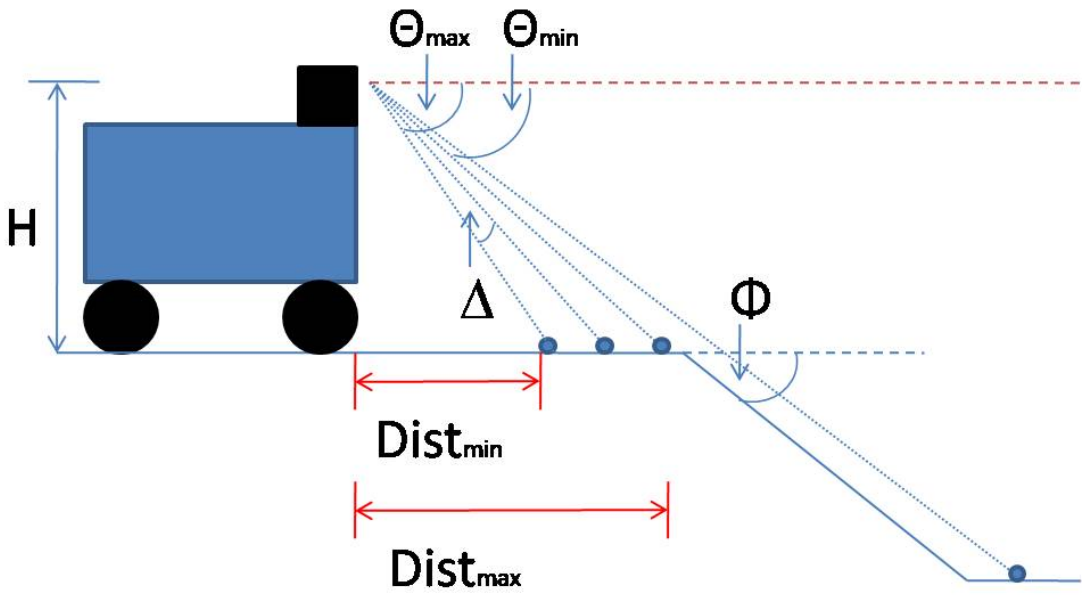
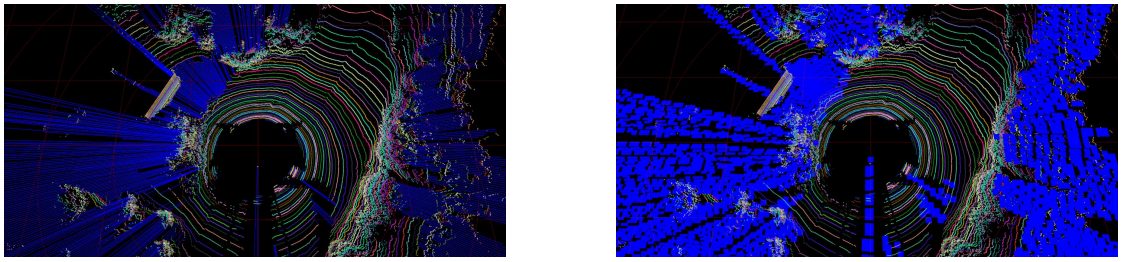


Figure 3.17: Sensor angles used to determine the min and max distance from the vehicle a steep negative decline can be discovered.

3.3.5 Translation to World Map

Those sections of the ray tracings that are classified as potential negative obstacles (Figure 3.18a) are translated into their world-model grid-cell counterpart and recorded as a hit of a potential negative obstacle (Figure 3.18b). If enough hits are recorded in a cell, it is classified as a potential negative obstacle and displayed on the map. Because negative obstacles are so difficult to detect at long ranges, this algorithm flags anything that might possibly be identified as a negative obstacle. Many false positives will thus be reported, and potential negative obstacles should not be treated as untraversable. Instead the robot is allowed to approach these areas, but should do so with caution and reduced speed, which can be incorporated into the path planning cycle. This software will be integrated with SPAWAR Systems Center Pacific's Autonomous Capabilities Suite (ACS) architecture. ACS will use its fuzzy-logic planner [Sights 07] to place potential negative obstacles as obstacles in the range-abstraction regions in front of the vehicle with a fuzzy-set value of *Close* or *Not Close*, which according to the fuzzy associate memory rules, causes the robot to approach more slowly but not stop.



(a) Potential negative obstacle rays (in blue)

(b) Potential negative obstacle cells (in blue) showing conversion from rays to grid cells

Figure 3.18: Potential negative obstacle displays. This analysis uses the vertical lidar scan (ray tracing) to classify negative hazard areas.

3.4 Traversability Analysis

A cell receives a traversability score based on the existence of a positive obstacle, a negative obstacle, a step edge obstacle, the steepness of the slope of its surrounding neighbors, and the slope residual (roughness).

The residual of the plane-fit function is used to determine roughness of the area, which is added into the traversability score of the cell. The traversability score is an integer value ranging from 0 to 255. The highest value, 255, is given to those cells that have a prohibitive classification (true negative obstacle, positive obstacle, step edge, or slope that is too steep). The classification type is stored in a single byte of data (8 bits), where each bit represents a different classification. In this way it can represent multiple classifications. The classifications with the highest values are the most prohibitive. A cell begins as a horizontal cell with a classification value of 1 (if enough data points lie within its x,y coordinates to correctly classify it). If it is classified as a steep slope, it adds a value of 32, or a 1 in the 6th bit place (the byte is now represented as binary 00100001 or as hex 0x21). If a grid cell has multiple classifications, the traversability score receives the maximum value of them all. For the slope and step edge analysis, where it might not exceed the obstacle threshold but comes close to it, the traversability score receives a value that is in relation to the ratio to the threshold (e.g. a slope of 15 degrees for a threshold of 30 degrees is scored at 0.5 of the maximum 255 prohibitive score value).

After the cell has been given its traversability score based on obstacles, step

edge, slope, and slope residual (see Figure 3.19), the path planner searches for the safest path that brings the vehicle closest to the goal location.

3.5 Path Planning

Without any a priori information, the data collected from most lidar sensors mounted on a robotic vehicle will not provide adequate range to create a large enough map to plan optimal paths to a goal location of any significant distance. In the true off-road tests with scattered vegetation, the majority of the gathered data from the large lidar mounted on the large UGV was within 50m or less, which would be traversed in 5s or less for a large vehicle such as the Max ATV travelling at 24kph. These methods focus on the near-field path planning using an arc-based planner (Figure 3.20), similar to the methods in [Larson 06] as well as on the NASA Mars Rovers. These arcs are dependent on the vehicle velocity $v = \frac{r}{\beta}$ where β is the turning rate of the vehicle. Because of this, the faster the vehicle is driving, the fewer available arcs can be used for path planning, since it is not desirable for the vehicle to become unbalanced and possibly roll. This particular 3D path planning algorithm simulates placing the platform centered on the cells of the arc (starting at the sensor and moving outward) and averaging the traversability scores for all the cells the vehicle would run over. The traversability score for each cell should already be calculated based on the terrain found inside it: positive obstacles, step edges, et cetera. Furthermore, penalties can be added for tip-over and high-centering [Roan 10], based on the 3D terrain underneath the vehicle. Due to the inherent curvature of the arcs and the disconnect between the actual skid-steering capabilities of these platforms (such as rotating in place), there will be arcs chosen which provide the best route for the first section of the path but eventually curve into obstacles. Since the robot will be calculating the traversability arc multiple times a second and revising its arc selection just as quickly, more weight is given to the cells of the arc in the closer regions than those in the farther regions. Deviation from the desired arc (the predetermined path) is also added into the score and a penalty is added for traversing multiple empty spaces (to keep from going too far into the unknown). The algorithm calculates the

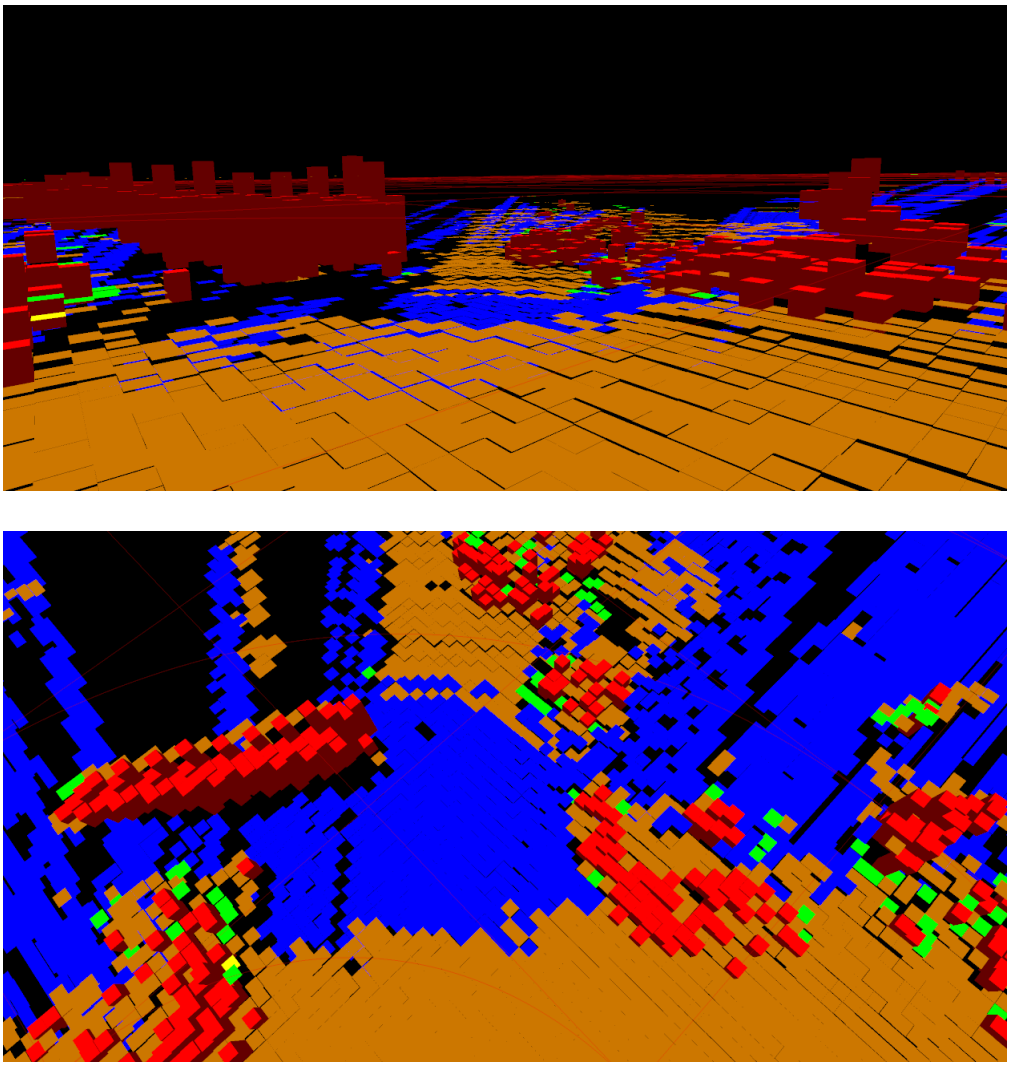


Figure 3.19: Side view and top-down view of grid cells painted on the screen with various colors representing the classification features: brown is a horizontal cell (ground), blue is a potential negative obstacle, green is a step edge, yellow is a steep slope, and red is a positive obstacle.

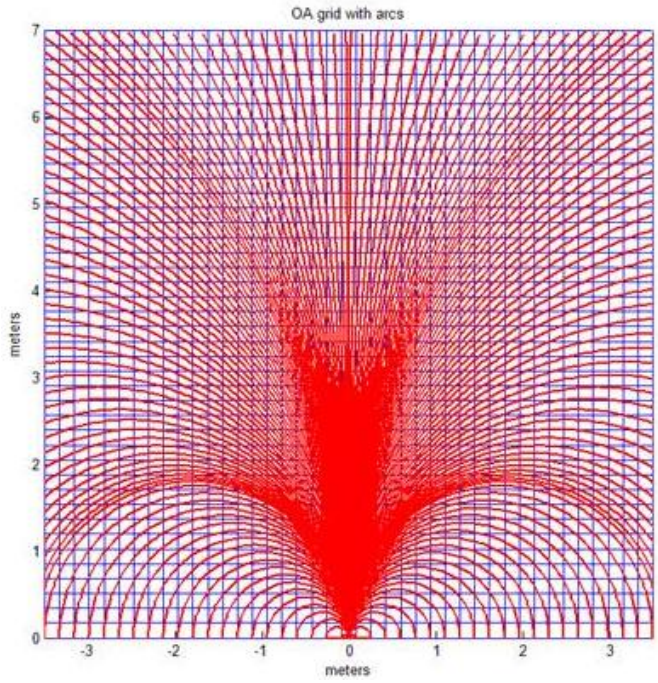


Figure 3.20: Path planning arcs. The arc with the best traversability score will be chosen

average traversability score as well as records the maximum score. If the maximum score exceeds a threshold of acceptability, the process is halted for this arc and the arc is discarded. The arc with the best score (including average traversability score, deviation score, roughness score, empty spaces score, and possibly tip-over or high-centerting score) is reported to the vehicle steering module (see Figure 3.21). For additional speed and reduced search, once an arc with a "good enough" score is found, it can be chosen as the desired arc and the search ends.

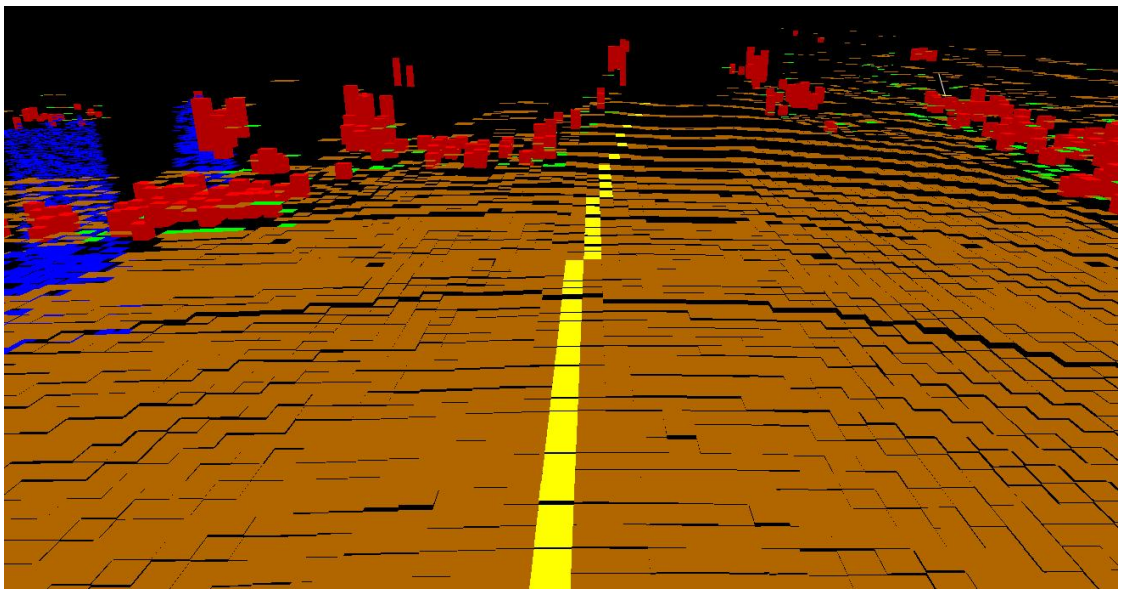


Figure 3.21: The chosen arc (yellow) after traversability scores have been calculated

Chapter 4

Experimental Evaluation

It is difficult to accurately label ground truth of negative obstacles in a real off-road environment. Multiple well-formed negative obstacles in one location are hard to come by in any natural setting, and they are very laborious to create. Given these hurdles, as well as limited funds and time, it was decided to measure performance of detection of negative obstacles in a simulated environment. In this way, a controlled environment can be created with ground truth of the exact number of negative obstacles, occluding positive obstacles, roughness of the terrain, and shape and spacing of the obstacles.

4.0.1 How to Measure Performance

The detection methods follow a vertical alignment of ladar beams and report the vector ray between two beams where it has detected a negative obstacle. For each negative obstacle in a simulated terrain map there will be multiple negative obstacle rays. Because each of the terrains were created from a simulation, the exact location and vector ray is known and is labeled as ground truth. If the detection methods finds a ray that overlaps one of the ground truth rays, it is considered a positive ray detection. Those detected rays that do not match up with a ground truth ray are considered a false ray detection. If there exists even one positive ray detection for a negative obstacle, this is counted as a positive obstacle detection. The results of negative obstacle ray detections as well as negative obstacle detections are reported in this paper.

4.0.2 Setup of Simulated Environment

To better analyze negative obstacles and their various representations, multiple simulated scenes were created, filled with negative obstacles that would cause damage or halt the robotic vehicle. These scenes were created by first building a height-map of the terrain with any number of negative and positive obstacles with specified sizes, as well any value of roughness for the rest of the terrain. The roughness was set by randomly selecting height values from a Gaussian distribution. To simulate the lidar returns from the terrain, a vector ray was created from the virtual location of the sensor on the height-map (depending on the size of the simulated platform), and a height-map intersection algorithm was used to simulate a lidar pulse from the sensor. This was performed by determining when the z value of the initial location of the lidar ray switches between "above" to "below" the terrain, or vice versa; comparing z values of the ray vector with the z values of the terrain heightmap at the same x, y location. The x, y, and z location where the lidar ray crosses over became the 3D point to be added to the point cloud list. The simulated 3D point cloud of the terrain was obtained by following the horizontal and vertical angular resolution pattern of both the small and large 3D lidar sensors that were used in this research (see Figure 4.1). The height-map for the smaller lidar had a width of 60m and length of 60m. Similarly, the height-map for the large lidar had a width of 200m and a length of 200m.

4.0.3 Selecting of Test Cases

For this experiment, the simulations set the height of the small lidar at 0.26m and the large lidar at a height of 1.81m because that is where they would be placed on the robotic vehicle platforms for this research.

The first two experiments were conducted simulating the small lidar mounted on the small UGV. The first test analyzed the NODR method only, with eight different terrains of varied negative obstacles and positive obstacles, four in smooth terrain and four in rough terrain. The second experiment included four tests on smooth terrain and four more on rough terrain, where the results were collected from both the NODR and the SVM methods. The average terrain height value for

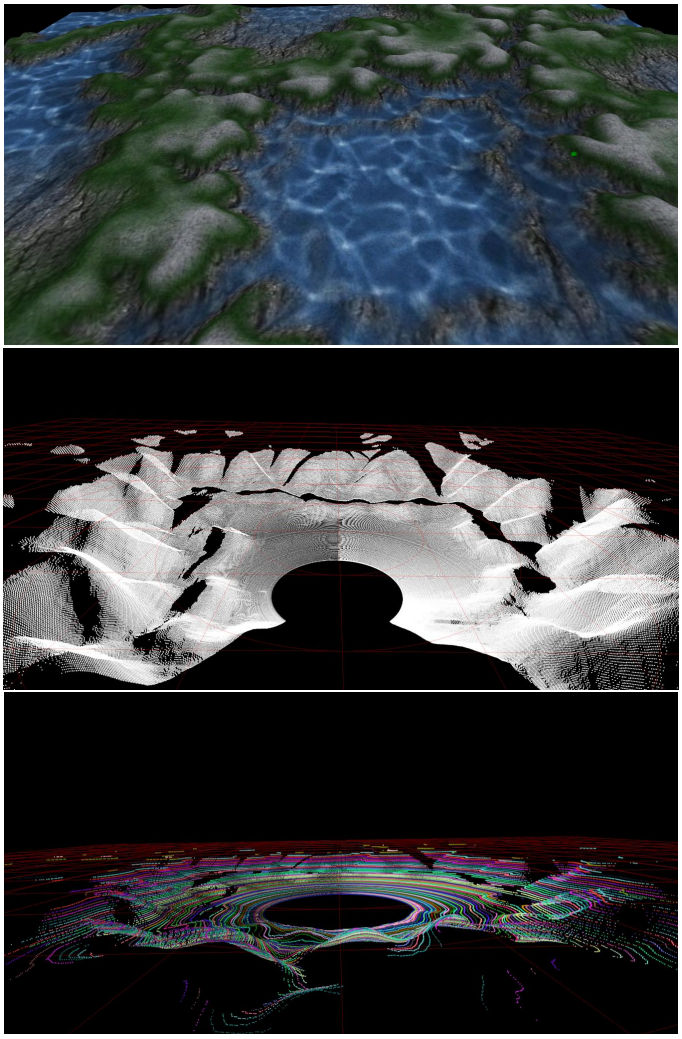


Figure 4.1: Simulated sample terrain (top) with lidar scans from both the small (middle) and large (bottom) 3D lidars

the small UGV was 0.5m, with a the variance of 0.0m for the smooth terrain and 0.0117m for the rough terrain. The negative obstacles for the small UGV terrain were each 0.5m deep, 0.5m wide, and 0.5m in length, about the minimum width or length or a negative obstacle that would still be a hazard to the small ground vehicle, while the positive obstacles were 0.5m tall.

The next two experiments were conducted simulating the large lidar mounted on a large UGV. The third experiment analyzed the NODR method only, with eight different terrains of varied negative obstacles and positive obstacles, four in smooth terrain and four in rough terrain. The fourth experiment includes four tests from smooth terrain and four more in rough terrain, collecting data from both NODR and SVM methods. The average height value for the terrain for the large UGV was 2.5m, with a variance of 0.0m for the smooth terrain and 0.0586m for the rough terrain. Some of these sample terrains included positive obstacles as well as negative. The negative obstacles for the large UGV terrain were each 2.5m deep, 1.0m wide, and 1.0m length, while the positive obstacles were 2.5m tall.

The SVM was trained with multiple sample height maps of equal amounts ground truth negative obstacle rays and non-negative obstacle rays, then tested with new simulated obstacle maps.

An experiment was also conducted in a true off-road environment. There were no ground truth negative obstacles or rays for this data set, but maximum distances for each detected classification class were measured.

4.0.4 Results of the Small Lidar Experiment (NODR Only)

Overall, the NODR method detected 52% of the negative obstacle rays through both smooth and rough terrain with a graph depicting the total number of obstacle rays detected vs. ground truth in Figure 4.2. Because of the assumption that if one negative obstacle ray is detected inside a negative obstacle, the whole obstacle is considered detected, the NODR method detected 78% of the negative obstacles (see Figure 4.3).

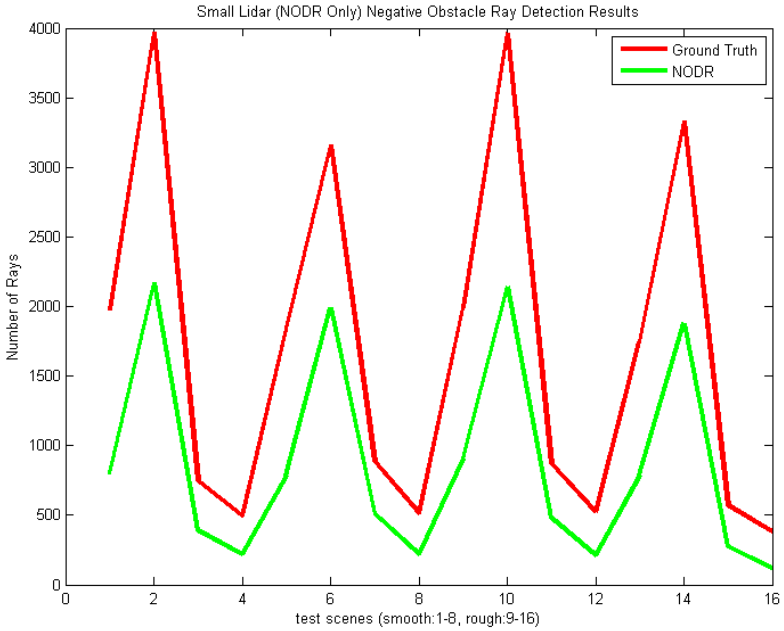


Figure 4.2: Negative obstacle rays detected correctly from NODR Only small lidar experiment

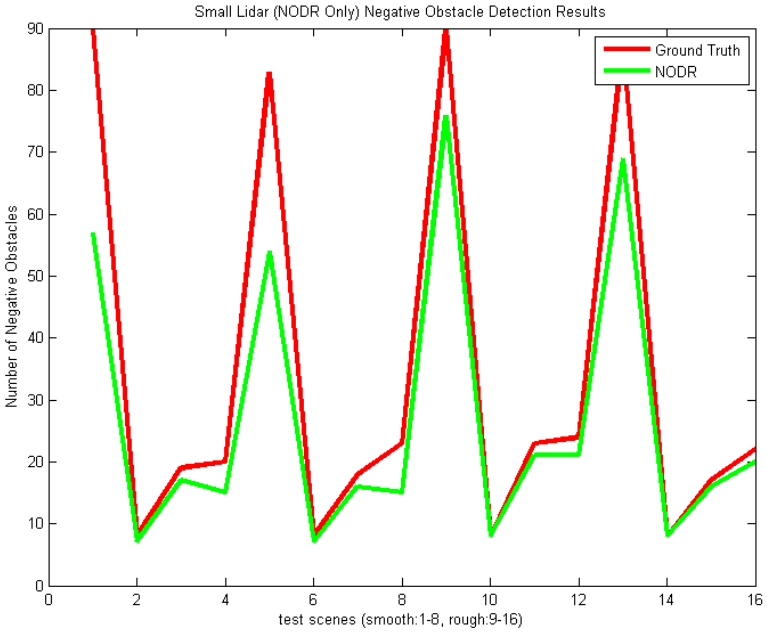


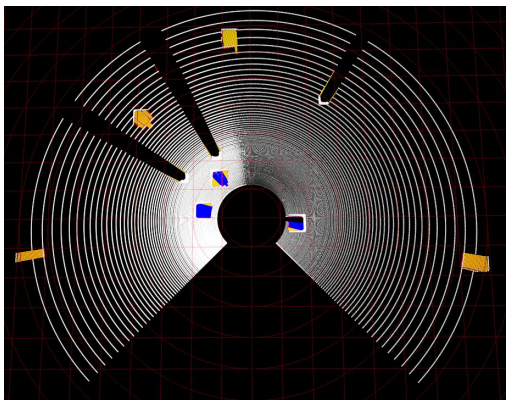
Figure 4.3: Negative obstacles detected correctly from NODR Only small lidar experiment

4.0.5 Results of the Small Lidar Experiment (NODR and SVM)

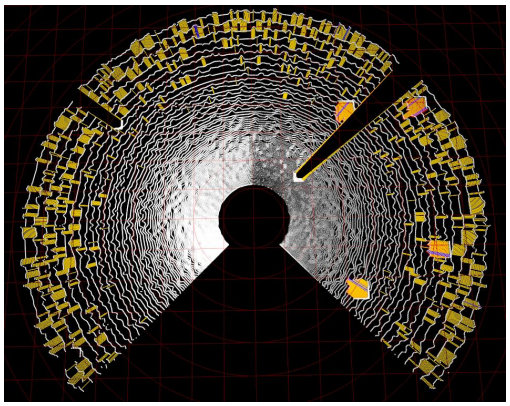
The SVM trained on 1714 smooth terrain samples and the 1682 rough terrain samples. In a very limited range of 8m, the SVM method detected 98% of the negative obstacle rays and 100% of the negative obstacles while the NODR only detected 39% of the obstacle rays and 78% of the negative obstacles. Figure 4.5 shows the number of correctly detected negative obstacle rays for the eight tests run for this particular sensor (four from the smooth terrain and four from the rough terrain) and Figure 4.6 shows the number of correctly detected negative obstacles. There is a tradeoff, however, because as the terrain becomes rougher, the reported false positives from the SVM method also increase. In all of the experiments from the smooth terrain, the SVM is accurate out to around 8m radius. However, in rough terrain, the SVM is only accurate out to around 6m radius. Experiments were conducted out beyond 6m in rough terrain and beyond 8m in smooth terrain and the SVM method innacurately identified every single ray beyond the 6m or 8m mark respectively as negative obstacles. The hypothosis is that the negative obstacles begin to be smaller than the length between vertical lidar pulses, which is about 0.5m for the smaller UGV platform. There might also be modifications in the selection of parameters that would increase the distance the SVM could accurately perform detection. The point cloud representation with negative obstacle detection images of the both the rough and smooth test cases are depicted in Figure 4.4.

4.0.6 Results of the Large Lidar Experiment (NODR Only)

Overall, the NODR method detected 27% of the negative obstacle rays through both smooth and rough terrain with a graph depicting the total number of obstacle rays detected vs. ground truth in Figure 4.7. Overall, the NODR method detected 31% of the negative obstacles and actual detection numbers for each terrain map is shown in Figure 4.8.



(a) Point cloud and detection results for the smooth NODR and SVM experiment for the small lidar



(b) The point cloud and detection results for the rough NODR and SVM experiemnt for the small lidar

Figure 4.4: Detection images for the smooth and rough NODR and SVM experiments for the small lidar. The gaps in data are the negative obstacles, and most are colored by the detection method (NODR is blue and SVM is yellow). The colors are layered, blue is on top, yellow below that, and red (for ground truth) is under that so that some colors are not visible in this image. There are positive obstacles as well, and all the data behind it is occluded from the lidar.

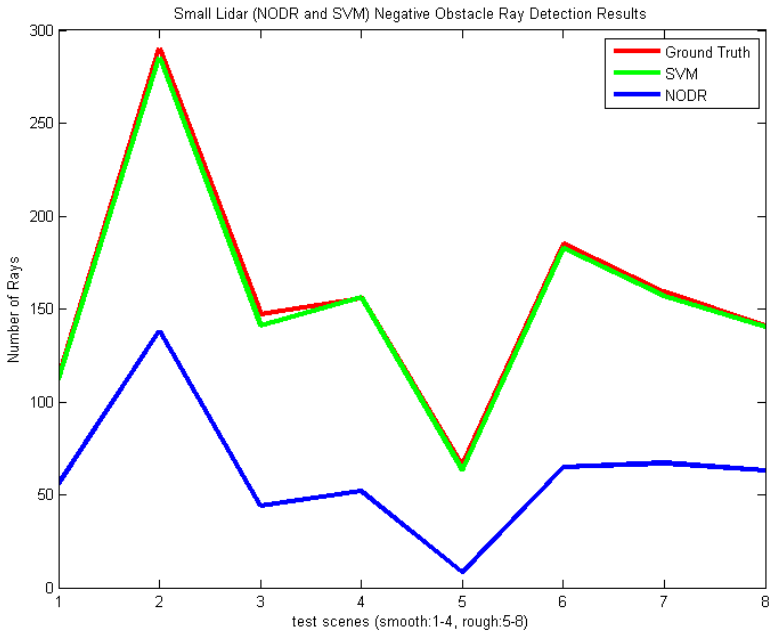


Figure 4.5: Negative obstacles rays detected correctly from SVM and NODR small lidar experiment

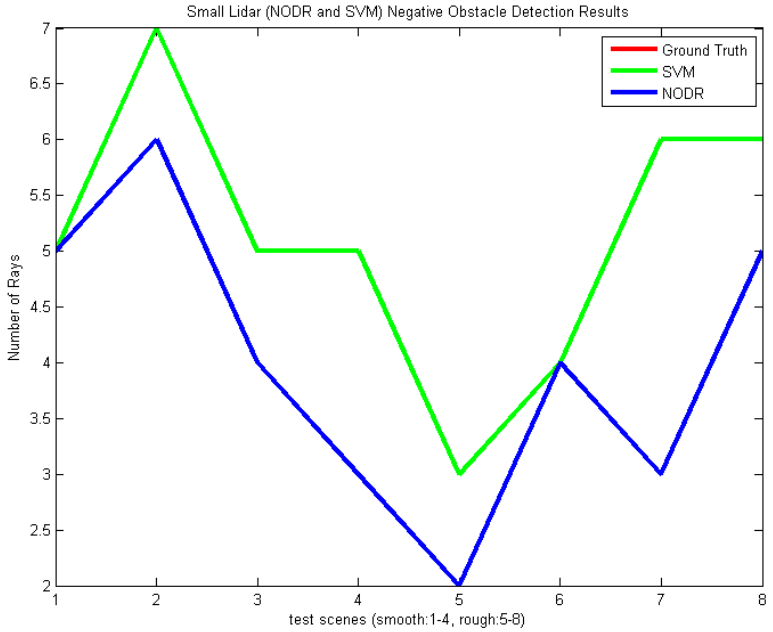


Figure 4.6: Negative obstacles detected correctly from SVM and NODR small lidar experiment. The red ground truth line is hidden behind the gree svm line which is 100% accurate.

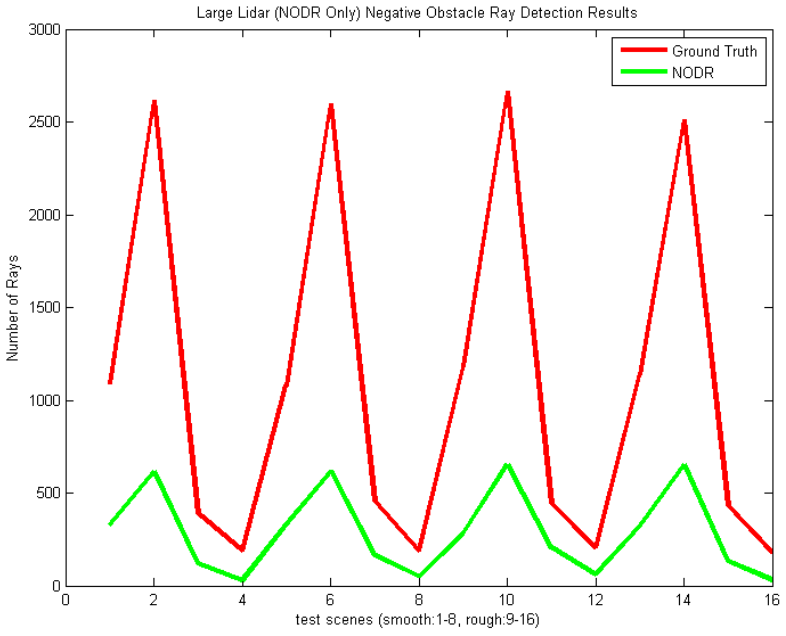


Figure 4.7: Negative obstacles rays detected correctly from NODR Only large lidar experiment

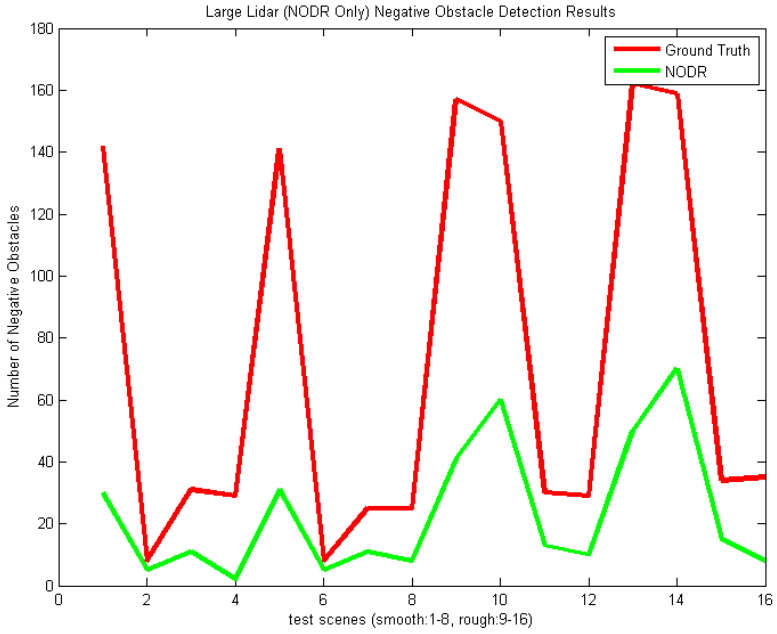
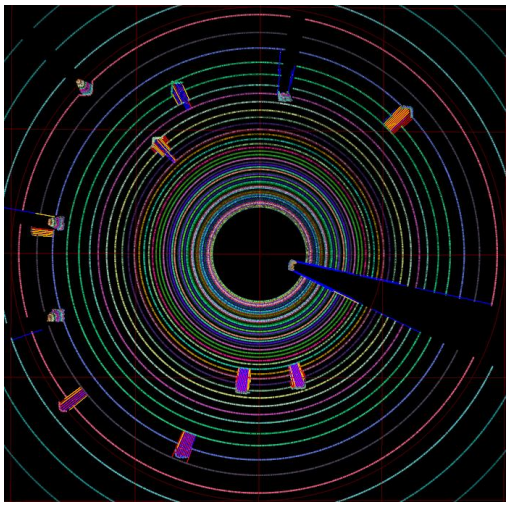


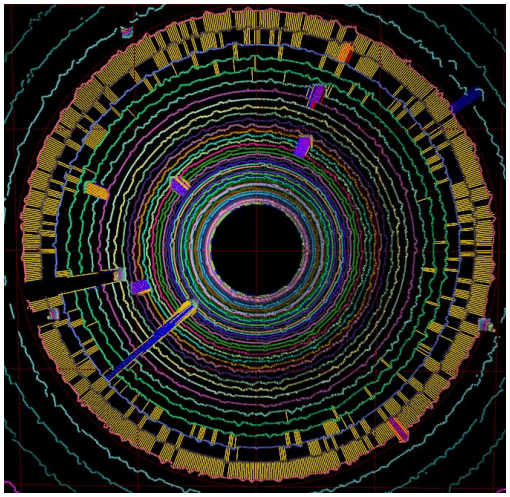
Figure 4.8: Negative obstacles detected correctly from NODR Only large lidar experiment

4.0.7 Results of Large Lidar Experiment (NODR and SVM)

The SVM trained on 1404 smooth terrain samples and 722 rough terrain samples. The SVM method detected 82% of the negative obstacle rays and 100% of the negative obstacles while the NODR method only detected 53% of the negative obstacle rays and 89% of the negative obstacles. Figure 4.11 shows the results from all eight tests run for this particular sensor (four from the smooth terrain and four from the rough terrain) and Figure 4.12 shows the number of correctly detected negative obstacles. As in the experiment with the small lidar SVM and NODR, there were increased amount of false positives for the rough terrain. In all of the experiments from the smooth terrain, the SVM is accurate out to around 20m radius. However, in rough terrain, the SVM is only accurate out to around 16m radius. Experiemnts were conducted out beyond 20m with the smooth terrain and 16m in the rough terrain and the SVM method innacurately identified most of the rays beyond the 20m and 16m mark respectively as negative obstacles, which can be seen in Figure 4.10. The point cloud representation with negative obstacle detection images of both the rough and smooth test cases are depicted in Figure



(a) Point cloud and detection results for the smooth NODR and SVM experiment for the large lidar



(b) Point cloud and detection results for the rough NODR and SVM experiment for the large lidar

Figure 4.9: Detection images for the smooth and rough NODR and SVM experiments for the small lidar. The gaps in data are the negative obstacles, and most are colored by the detection method (NODR is blue and SVM is yellow). The colors are layered, blue is on top, yellow below that, and red (for ground truth) is under that so that some colors are not visible in this image. There are positive obstacles as well, and all the data behind it is occluded from the lidar.

4.9.

4.0.8 Results of Real Off-road Course

Data was also collected on a true off-road coarse in the coastal areas in Point Loma, CA. An image of the course and the route taken can be seen in Figure 4.13. The lidar collected 1122 frames of point cloud data, and this system was able to process each frame, detecting obstacles and providing a suggested route, as well as display the data on the screen, for the front half of the data (everything in front of the vehicle), at an average rate of 2.23Hz on a dual-core laptop. The maximum detected ranges of obstacle features in this data set are provided in table 4.1.

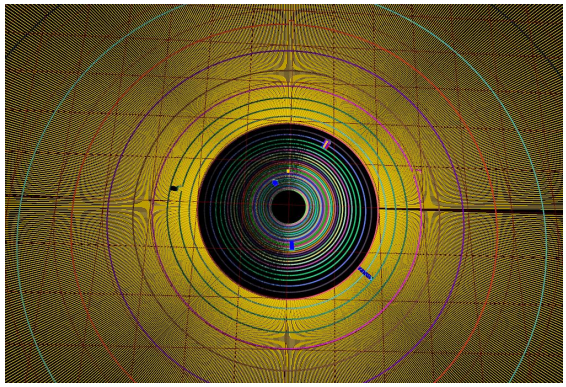


Figure 4.10: False positives (yellow) reported for ranges that were beyond 16m for the simulated large lidar sensor

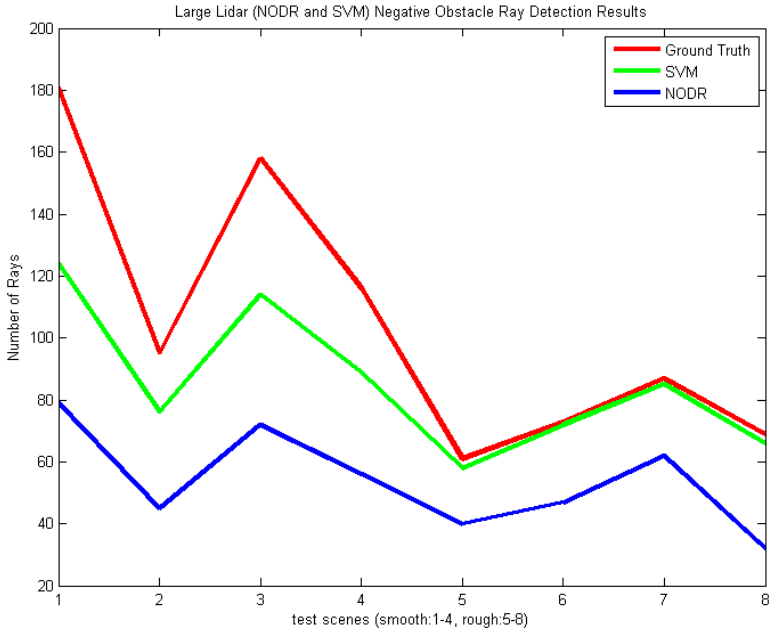


Figure 4.11: Negative obstacles rays detected correctly from SVM and NODR large lidar experiment

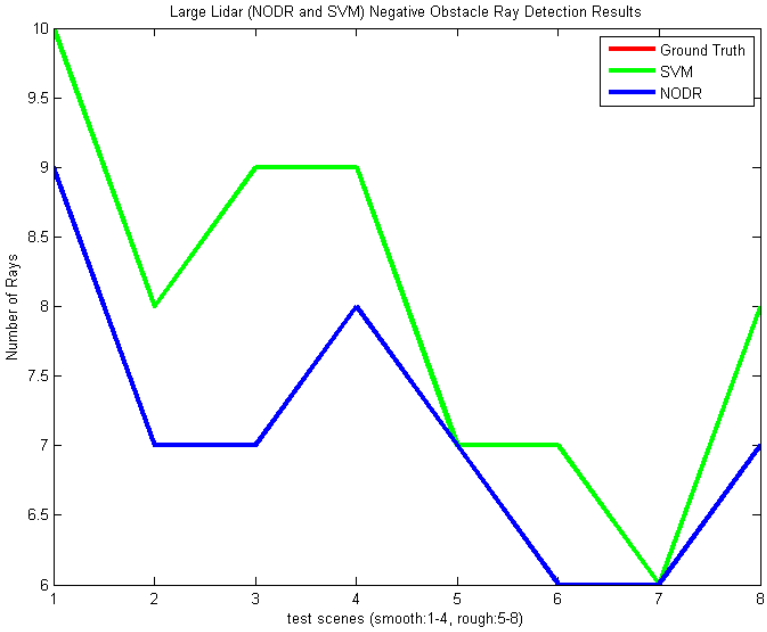


Figure 4.12: Negative obstacles detected correctly from SVM and NODR large lidar experiment. The red ground truth line is hidden behind the green svm line which is 100% accurate.

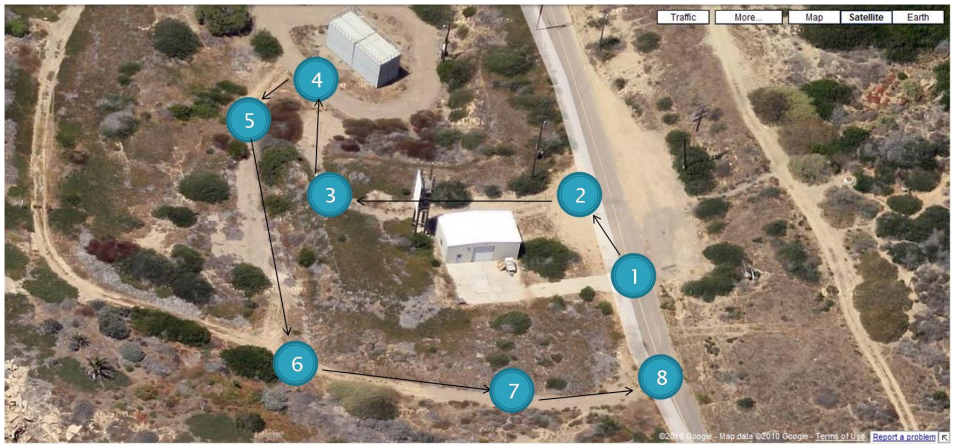


Figure 4.13: Google Maps satellite image of the off-road course with numbers ordered to show the route taken

Table 4.1: Maximum detection ranges of obstacles found in off-road test route

Obstacle Feature	Maximum Detection Range	Description
Positive obstacle	113.4	Vegetation
Negative obstacle	82.6	Steep slope
Overhang	100.6	Tall vegetation
Step edge	115.1	Steep hill
Steep slope	115.1	Steep hill

Chapter 5

Conclusion and Future Work

Based on the results from these experiments and according to the stopping distance equations, the small vehicle can travel 2.5m/s, which is its maximum speed, process the data in 0.5s, and still stop 2.2m away, far away from the SVM rough terrain range limit of 6m. As well, the large vehicle can travel 37kph, process the data in 0.5s, and stop 14m away, just shy of the svm rough terrain range limit of 16m. This is a good benchmark to start from for the goal of high-speed off-road autonomous driving.

The best solution for negative obstacle detection is a combination of SVM for the short range and NODR for long range detection. Understanding this exact range limit and how to push it out even further is a task for future work and could provide even better results.

It has been demonstrated that it is possible to analyze terrain data, classifying hazards such as positive obstacles, steep slopes, step edges, and even negative obstacles at a distance far enough to travel at relatively high-speeds. It is possible to produce traversability scores for the terrain and plan traversable paths for a small and large UGV off-road environments, all in near-real-time using only a single 3D lidar sensor. The framework has been provided for analyzing the 3D geometry of the terrain during path planning to begin looking for paths that will avoid tip-over and high-centering. This work will be beneficial for autonomous UGV military and commercial applications in off-road terrain.

The traversability analysis software created from this research has been tested on both a large and a small UGV platform with two different lidar sensors that

use very different techniques for 3D sensing. It can be assumed that this software can be ported to a variety of UGV platforms, and has application for other sensors that create 3D point clouds.

One of the issues of attempting to detect negative obstacles at long ranges is the vertical angular resolution of the 3D lidar. A possible solution to improving this angular resolution is to retain multiple scans and analyze them as the vehicle is moving, either from the 3D lidar or even a single-scan lidar, angled in such a way that it would find negative obstacles far enough away to react in time. The distance between consecutive scans would be dependent only on the refresh scan rate of the lidar and the speed of the vehicle. For instance, a lidar scanning at 40Hz on a vehicle traveling at 32kph can detect the ground every 0.22m (which can be set any distance away from the vehicle). Currently the smaller lidar skips 2m between lidar scans at 16m away. The large lidar skips 32m between lidar scans 75m away. This could provide better accuracy for negative obstacle detection.

A major classification class that has not yet been explored in this research is vegetation. Most off-road terrain is littered with vegetation that, to a simple obstacle detector, appears to be an obstacle. The current path planner will not allow the unmanned vehicle to select narrow pathways that would scrape the vegetation. Vegetation detection could be done by analyzing the visible spectrum from a camera and combining that with lidar data for enhanced understanding of the environment; by clustering the data points and classify scatter, linear, or planar objects [Lalonde 06]; or through analysis of multi-return lidar.

Bibliography

[Barnard 02] K. Barnard, L. Martin, A. Coath & B. Funt. *A comparison of computational color constancy Algorithms. II. Experiments with image data*. IEEE Transactions on Image Processing, vol. 11, pages 985–996, September 2002.

[Belongie 04] Serge Belongie. *Lecture 10 Camera Calibration*. 2004.

[Carsten 06] J. Carsten, D. Ferguson & A. Stentz. *3D Field D*: Improved Path Planning and Replanning in Three Dimensions*. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 3381–3386, October 2006.

[Congress 01] United States Congress. *Report to Congressional Committees: Army Transformation Faces Weapon System Challenges*. 2001.

[Fong 03] E. Fong, W. Adams, F. Crabbe & A. Schultz. *Representing a 3-d environment with a 2 1/2-d map structure*. In Intelligent Robots and Systems (IROS 2003), volume 3, pages 2986–2991, October 2003.

[Halterman 10] R. Halterman & M.H. Bruch. *Velodyne HDL-64E lidar for unmanned surface vehicle obstacle detection*. In SPIE Proc. 7692: Unmanned Systems Technology XII, pages 644–651, Orlando, FL, April 2010.

[Heckman 07] N. Heckman, J-F. Lalonde, N. Vandapel, & M. Hebert. *Potential Negative Obstacle Detection by Occlusion Labeling*. In Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2007.

[Holste 09] S. Holste & D. Ciccimaro. *Increasing the mobility of dismounted marines*. Technical Report 1988, SSC Pacific, 2009.

[Hong 98] T. Hong, S. Legowik & M. Nashman. *Obstacle Detection and Mapping System*. National Institute of Standards and Technology (NIST) Technical Report NISTIR 6213, pages 1–22, 1998.

[Lalonde 06] J-F. Lalonde, N. Vandapel, D. Huber & M. Hebert. *Natural Terrain Classification using Three-Dimensional Ladar Data for Ground Robot Mobility*. Journal of Field Robotics, vol. 23(10), pages 839–861, November 2006.

[Larson 06] J. Larson, M. Bruch & J. Ebken. *Autonomous Navigation and Obstacle Avoidance for Unmanned Surface Vehicles*. In Proceedings of SPIE Unmanned Systems Technology VIII, Orlando, FL, April 2006.

[Manduchi 04] R. Manduchi, A. Castano, A. Talukder & L. Matthies. *Obstacle Detection and Terrain Classification for Autonomous Off-road Navigation*. Autonomous Robots, vol. 18, pages 81–102, 2004.

[Matthies 94] L. Matthies & P. Grandjean. *Stochastic Performance Modeling and Evaluation of Obstacle Detectability with Imaging Range Sensors*. IEEE Transactions on Robotics and Automation, vol. 16(12), December 1994.

[Matthies 03] L. Matthies & A. Rankin. *Negative Obstacle Detection by Thermal Signature*. In Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 906–913, 2003.

[Murarka 08] A. Murarka, M. Sridharan & B. Kuipers. *Detecting Obstacles and Drop-offs using Stereo and Motion Cues for Safe Local Motion*. In International Conference on Intelligent Robots and Systems (IROS), 2008.

[Rankin 09] A. Rankin, A. Huertas & L. Matthies. *Stereo Vision Based Terrain Mapping for Off-Road Autonomous Navigation*. In Proceedings of SPIE, volume 7332, Orlando, FL, 2009.

[Roan 10] P. Roan, A. Burmeister, A. Rahimi, K. Holz & D. Hooper. *Real-World Validation of Three Tipover Algorithms for Mobile Robots*. In IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, May 2010.

[Seraji 03] H. Seraji. *Rule-Based Traversability Indices for Multi-Scale Terrain Asessment*. In Proceedings of 2003 IEEE International Conference on Control Applications, Istanbul, Turkey, June 2003.

[Shoemaker 98] C.M. Shoemaker & J.A. Bornstein. *The Demo III UGV program: a testbed for autonomous navigation research*. In IEEE International Symposium on Intelligent Control, pages 644–651, Gaithersburg, MD, September 1998.

[Sights 07] B. Sights, G. Ahuja, G. Kogut, E.B. Pacis, H.R. Everett, D. Fellars & S. Hardjadinata. *Modular Robotic Intelligence System Based on Fuzzy Reasoning and State Machine Sequencing.* In SPIE Proceedings 6561: Unmanned Systems Technology IX, Defense and Security Symposium, Orlando, FL, April 2007.

[Silver 06] D. Silver, B. Sofman, N. Vandapel, J. Bagnell & A. Stentz. *Experimental analysis of overhead data processing to support long range navitation.* In International Conference on Intelligent Robots and Systems, 2006.

[Simmons 95] R. Simmons & et al. *Experience with rover navigation for lunar-like terrains.* In Proceedings of Conference on Intelligent Robots and Systems, Pittsburgh, PA, August 1995.

[Thrun] Sebastian Thrun. *What we're driving at (Blog Post).*

[Thrun 06] Sebastian Thrun, Michael Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, Kenny Lau, Celia Oakley, Mark Palatucci, Vaughan Pratt, Pascal Stang, Sven Strohband, Cedric Dupont andLars Erik Jendrosseka, Christian Koelen, Charles Markey, Carlo Rummel, Joe van Niekerk, Eric Jensen, Philippe Alessandrini, Gary Bradski, Bob Davies, Scott Ettinger andAdrian Kaehler, Ara Nefian & Pamela Mahoney. *stanley: the robot that won the darpa grand challenge.* Journal of Field Robotics, 2006.

[Triebel 06] R. Triebel, P. Pfaff & W. Burgard. *Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing.* In Proceedings of the International Conference on Intelligent Robots and Systems, IROS, 2006.

[USMC 11] USMC. *Information Paper.* Technical report, Marine Corps Warfighting Laboratory in Quantico, VA, April, 2011.

[Vapnik 95] V. Vapnik & C. Cortes. *Support-vector networks.* Machine Learning, vol. 20, pages 273–297, 1995.