

CLASSIFYING GROUND TERRAIN USING MULTIVIEW METHODS

BY ELIE ROSEN

A thesis submitted to the
Graduate School—New Brunswick
Rutgers, The State University of New Jersey
in partial fulfillment of the requirements
for the degree of
Master of Science
Graduate Program in Electrical and Computer Engineering

Written under the direction of

Dr. Kristin J. Dana

and approved by

New Brunswick, New Jersey

May, 2015

ABSTRACT OF THE THESIS

Classifying Ground Terrain Using Multiview Methods

by **ELIE ROSEN**

Thesis Director: Dr. Kristin J. Dana

In this thesis, we acquire outdoor and indoor ground terrain photos at multiple viewing angles in natural light as a means to discover fundamental differences between terrains due to reflectance variation in natural illumination. Ground terrain affects robot control algorithms but little attention has been given to automatically determining the ground terrain composition using vision-based recognition techniques. For example, autonomous cars that can understand the presence of ice and wet road surfaces can automatically determine potential hazards, this in turn allows the car to adjust its driving accordingly to keep passengers safe. We examine an existing classification method, Texton Boost, to set the basis for comparison to a new proposed method based on angular gradient differential illumination histograms for automatically determining and classifying ground terrain surfaces. A database of 50 ground covers, each imaged from 33 viewing angles including finely quantized angular directions for computing gradients, was acquired by a mobile robot platform for this work. A selection of terrains in the database include gravel, carpet, tile, snow, ice, pebbles, grass, leaves, concrete, and asphalt. Multiple outdoor samples also include wet and dry varieties. The database of 1,650 ground terrain images is made publicly available for research use.

Acknowledgements

I would like to thank my advisor Dr. Kristin Dana for all of her support, advice, and guidance throughout the the many stepping stones to the culmination of my thesis. Her guidance over the past years has been enormous part in shaping who I am as a researcher today. I also would like to extend my gratitude to my committee members Dr. Ivan Marsic and Dr. Saman Zonouz. Lastly, I would like to thank my colleagues in the computer vision lab; Hang Zhang for his help and expertise in Texton Boost, Eric Wengrowski, Parneet Kaur, and Jennifer Shane. A special thanks goes to Gradeigh Clark for opening doors that I could not open myself.

Table of Contents

Abstract	ii
Acknowledgements	iii
List of Tables	vii
List of Figures	viii
1. Introduction	1
1.1. Background	5
2. System Design	8
2.1. Pioneer 3-AT Mobile Robot	8
2.2. Cyton Gamma 300 Robot Arm	9
2.3. Basler Ace Area Scan Camera	11
2.3.1. Lens selection methodology	12
2.4. Software and Central Processing	13
3. Computer Vision and Robot Integration	17
3.1. Design Considerations	17
3.2. Communication	19
3.3. System Coordinate Frame	19
3.3.1. Maximum sample area	21
3.3.2. Smallest feature size	21
4. Ground Terrain Database	26
4.1. Capture Point locations	26
4.1.1. Small angular changes for gradient computation	27

4.2. Camera Pose	27
4.3. Manipulation Action Sequence	28
4.4. Collection of Samples	30
4.4.1. Camera gain, white balance, and exposure	30
4.4.2. Digital storage requirements	32
5. Classifying Ground Terrain Surfaces	34
5.1. Image Preprocessing	35
5.2. K-means based Texton Learning	35
5.3. Dense Texton Mapping	36
5.4. Texton Boost	37
5.5. Weighted Histograms	38
5.6. Classification	38
5.6.1. Single Angle Histograms	39
5.6.2. Eleven Angular Gradient Histograms	39
5.6.3. Pair-wise Angular Gradient Histograms	39
6. Results	40
6.1. Angular Gradient Differential Histogram Classification	40
6.1.1. Eleven angle histograms	40
6.1.2. Single angle histograms	40
6.1.3. Pairwise angle histograms	41
6.2. Texton Boost Performance	44
6.3. Discussion	45
7. Conclusion and Future Work	48
Appendix A. Table of Imaging Poses	50
Appendix B. Code for Automated Camera Control and Arm Movement	52
Appendix C. Imaging Procedure	57

Appendix D. List of Database Samples	58
References	61

List of Tables

2.1. Specifications of the Pioneer 3-AT Mobile Robot [2]	14
2.2. Specifications of the Cyton Gamma 300 Arm [77]	15
2.3. Specifications of the Basler Ace Area Scan Camera [8]	16
2.4. Specifications of the TechSpec High Resolution Lens	16
6.1. Recognition rates for different combinations of pairwise angle histogram classification (mean only)	43
6.2. Recognition rates for multiple methods of pairwise angle histogram clas- sification	43
6.3. Histogram intersection for wet and dry terrains	47
A.1. Position and poses for camera placement. Arm positions with respect to the arm base are postfixed with a . Gripper positions with respect to the surface coordinate frame are postfixed with b	51
D.1. List of all collected samples and labels	60

List of Figures

1.1. Mobile robot based sample collection platform	2
1.2. Comparison between dry and wet images of concrete captured in natural light. Wet concrete has reflective properties that are not present in dry concrete.	6
1.3. Comparison of different types of carpet textures and materials	7
1.4. Comparison of common outdoor terrains	7
1.5. Comparison of cold weather related terrains	7
2.1. Pioneer 3-AT Mobile Robot	9
2.2. Mobile robot power distribution diagram	10
2.3. Cyton Gamma 300 servo power distribution	10
2.4. Cyton Gamma 300 Robot Arm	11
2.5. Basler Ace <i>acA2040-90uc</i>	12
2.6. TechSpec 25mm Camera Lens	13
3.1. Complete robot system design	18
3.2. Mobile robot network design	20
3.3. Coordinate frame transformation from arm coordinate frame to sample area coordinate frame. The sample area origin is denoted as a small sphere.	23
3.4. Maximum arm reach for a radius of 304.8 mm from sample area. Red borders show the furthest points the robotic arm can reach forming a partial hemisphere of coverage over a sample.	24
3.5. Field of view as a result of camera working distance	25
4.1. The process of taking samples follows this pipeline, positions are loaded, poses are set, and finally a manipulation action sequence occurs. This repeats for every sample.	27

4.2. Top down view of capture point locations	28
4.3. Arm placed in position <i>point4_1</i>	29
4.4. Sequence of arm movement from <i>point2_3</i> to <i>point3_1</i> by waypoint <i>point9_1</i>	30
4.5. Example of arm movement failure moving from <i>point2_3</i> to <i>point3_1</i> . .	31
4.6. Robot setup to capture a ground terrain sample outdoors	32
5.1. High level overview of terrain classification process	35
5.2. Process for converting training images to texton dictionary	36
5.3. Process for converting training images to dense texton maps	37
5.4. Select results from the texton map training	37
5.5. Weighted histograms for multiple ground terrains	38
6.1. Confusion matrix for eleven angle histogram classification. Overall recog- nition rate is 98%	41
6.2. Confusion matrix for single angle histogram classification. Overall recog- nition rate is 88.42%	42
6.3. Confusion matrix for pairwise angle histogram classification. Overall recognition rate is 89.45%.	44
6.4. Confusion matrix of Texton Boost performance in image form, overall recognition rate of 34 unique terrains is 83.68%	45
6.5. Comparison of 100% sample recognition between single angle and pair- wise classification. Single angle has 15 instances of 100% recognition, pairwise has 24 instances of 100% recognition.	46
6.6. Texton histograms for ice and snow/ice mixture terrains (<i>histogram_intersection</i> = 95%). The histogram for sand is also added to provide further compari- son to an additional sample.	46
6.7. Incorrect classification of the wetdirt3 sample. Here we see incorrect labeling to wet dirt, wet and dry pavement, and pine needles.	47
6.8. Comparison of dry and wet terrain histograms	47
7.1. Comparison of cold weather related texton maps	48
7.2. Comparison of texton maps for wet and dry samples of concrete	49

Chapter 1

Introduction

In recent years there has been a steady increase of technical advances in autonomous navigating robots, unmanned aerial vehicles, and self driving cars. A recent study on self driving cars even proclaims that by the year 2030, about 20% of all vehicles on the road will have full autonomous capabilities [54]. Currently these systems rely mostly on physical measurement sensors such as LIDAR for detecting the location of roads, obstacles, and intersections, and cameras for detecting road signs and lane markers on pavement [38]. However, the largest problem facing wide-scale adoption of these vehicles depends on the cars ability to drive safely in *all* driving conditions including more common environmental conditions such as rain and less common but more dangerous weather such as snow and ice [52]. In this thesis, we present methods for automatically detecting and classifying ground terrains through image-based methods, we assert that terrains can be classified on the basis of reflectance alone. An autonomous vehicle that is aware of its local terrain can be integrated with better navigation algorithms to avoid potential dangerous situations. Additionally, general mobile robotic devices including indoor robots can benefit from knowing the terrain they traverse.

To facilitate this work, we require a large collection of sample terrains in a multitude of settings including outdoors and indoors along with outdoor attributes of both wet and dry surfaces. A novelty of this work includes removing the lab setting from sample collection. Within a lab, too many characteristics of typical sampling procedures attempt to control variables such as lighting, camera settings, and material quality such as non-uniform or ideal surfaces that contain debris. These methods of collection are too idealized and fail to give adequate representations of real materials in real world environments. Examples of this can be found in [42, 51, 84] Instead, we have taken the

lab outdoors or rather “into the wild” [16]. We designed a mobile robot platform for collecting terrain samples in their natural environment and undisturbed from human interaction. The robot in Figure 1.1 is our mobile lab. Here we work with non-ideal lighting conditions that are determined by the time of year, time of day, and clarity of the sky. By bringing the lab outdoors we are able to take samples of natural environment phenomena such as snow and ice, we also can examine differences in materials affected by rain and small particle erosion as discussed later on in this chapter.



Figure 1.1: Mobile robot based sample collection platform

Material recognition through reflection is not new, there are many examples that span the last decade of ground breaking work in this area [24, 26, 51, 39, 25, 44, 55]. Some work has been done in the space of classifying terrains through physical means by measuring the friction experienced while navigating over terrain [80, 5, 76] however, the classification of terrains and surfaces do not occur until after the robot has passed over the terrain thus making this solution not possible for the use cases of safe autonomous

navigation. While there are also many examples of terrain recognition out in the field [102, 11, 45, 108, 14, 43], none explore the use of purely reflective and illumination based methods. This work, examines the classification of ground terrains through only image based methods.

To form our solution to this problem, we look at previous work specifically relating to texture recognition with textons and texton histograms. Textons [51] are defined as fundamental micro-structures that represent texture within an image. By placing images through a filter bank, we can find clusters of texture features. A single cluster of texture features defines a texton. In a given image we can define as few or as many textons as we desire. Ultimately, the number of unique textons defined is a trade off between how different textons are from each other or over-fitting the data which would prevent the textons from being able to represent images with minor variations. A collection of textons is referred to as a texton dictionary. Once textons are determined across an image, we need to find an appropriate method for storing this information. In Cula 2001 [18] textons are stored in n -dimensional histograms where n is the number of textons represented. For a given image, each pixel is mapped to a texton to form a texton map. The sum of a particular texton across an image is then added to its representative bin in the histogram. Each histogram forms a fingerprint that uniquely represents the texture within an image. That is, for a particular material, we can create a material profile through a collection of histograms taken from multiple viewing angles and levels of illumination [19]. Some work has previously been done in the space of classifying materials without the need for a fixed viewpoint and illumination through use of rotationally invariant filters to define the texton dictionary [92]. There is also some work in the space of classification using only single images [94]. However, we can build a more robust classification through use of multiple viewing angles.

One such method for classifying materials is Texton Boost [85], the authors implement a system for automatic detection, recognition, and segmentation of object classes in photographs. Each segmentation in a photograph can then be classified to a material or object. The initial purpose for Texton Boost is for multiple object scenes but we use it to find dense clusters of similar texture across local samples of terrain materials. We

use these results as an initial benchmark of performance and implement a similar but novel method for classification using small angular gradient differential histograms. In this method, we collect images of terrain samples and combine features of illumination differences in photographs to build dense texton maps and weighted histograms based on small changes in capture point location.

For this work, we introduce a new approach to classifying terrain through use of small angular gradients between multiple viewpoints. We define angular gradients as the combination of textons between two viewpoints separated by a differential angle δ . This approach is unique compared to prior work that has been done based on large changes in angular viewpoints evenly distributed across a hemisphere with respect to the sample under test [40, 68, 71]. We find that these methods are too restrictive in that they require coarsely quantized camera positioning to accurately represent materials. Instead, in this work while also using a hemisphere image capture approach, we also capture additional viewing angles that are separated by a small delta to form angular gradient histogram pairs to build a more robust training set.

In combination with recent work such as [109], we are looking to explore the effects of light reflectance on surfaces that are not guaranteed to be homogenous in nature. Some examples of this include concrete with stones and snow that is mixed with road debris. We also examine terrains from a macro perspective to extract texture that defines overall physical properties of terrains.

This thesis is organized as follows, Chapter 1 motivates real world situations and examples for meaningful applications to this work. It also provides insights to current work in the field of material recognition and background knowledge to the effects of reflectance based illumination on materials. Chapter 2 provides details on the physical design of a mobile robot platform for acquiring ground terrain samples out of the lab and in the wild with all of the requirements and characteristics required to build a ground terrain sample database. Chapter 3 brings the individual components discussed in Chapter 2 all together to integrate into a full robotics platform that can automatically take samples and navigate. Chapter 4 discusses the intricacies of imaging samples of materials without the controlled environment of a lab to form a database of real

world terrain samples in their natural environment. This process includes a strategically planned positioning, posing, and sequential action sequence to coordinate the collection of samples at a terrain location. Chapter 5 discusses methods for automatically determining terrains. Chapter 6 discusses the results of these findings. Finally, in Chapter 7 we provide conclusion to this work and comment on the success of these methods and the future of work in this area of research.

1.1 Background

This research attempts to address two issues facing modern recognition of terrain surfaces. That is, the field of autonomous navigation lacks a proper collection of ground terrain samples of surfaces that can look very different depending on outdoor environmental conditions including weather and lighting. It also provides a classification method based on illumination.

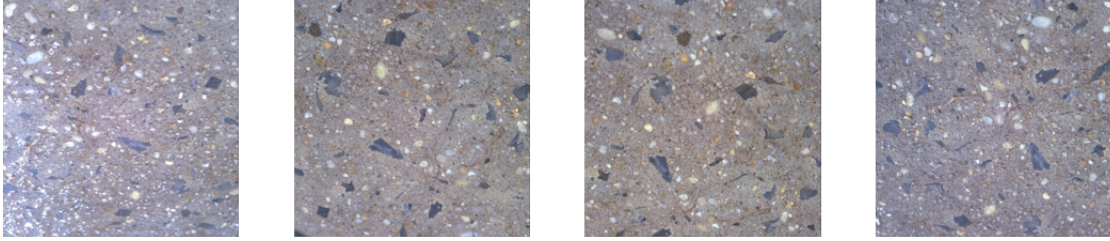
Consider for example the same material imaged in two separate environments, in Figure 1.2a-d we see a normal sample of concrete exposed only to natural sunlight. The concrete contains many small stones; however they blend together to form a terrain that is uniform in depth and fairly consistent in color due to evenly distributed dirt particles. Now examine Figure 1.2e-h; here we see the same composition of concrete however it is saturated with natural rain. The rain removes many of the surface dirt particles and causes the rocks to stand out and have sharper edges, you can also see additional artifacts of light reflection on many of the edges. Overall this sample can look drastically different under different conditions.

To further motivate the problem, take for example a series of indoor terrains as shown in Figure 1.3, here we see three different types of carpet illuminated by fluorescent lighting. We can see vast differences in texture and crevices where light is reflected.

Terrain images from outdoor environments tend to have less similarities. Take for example samples of dirt, grass, and pavement in Figure 1.4. Here we see three textures that while can be located close to each other spatially have textures that reflect light very differently.



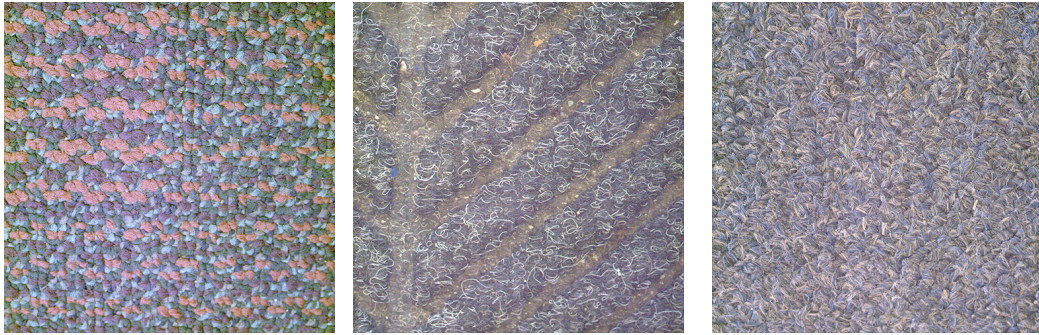
(a) Dry concrete captured at multiple angles



(b) Wet concrete captured at multiple angles

Figure 1.2: Comparison between dry and wet images of concrete captured in natural light. Wet concrete has reflective properties that are not present in dry concrete.

A very interesting part of this work is the examination of cold weather based phenomena such as snow, ice, and partially melted snow/ice mixtures as shown in Figure 1.5. Snow is a very clear uniform white but also contains many specularities throughout the sample, this feature is unique only to snow within the completed set of samples. Ice is also interesting since it behaves almost like a mirrored surface. Melted snow is very opaque in nature and tends to absorb light which causes it to appear darker and is less reflective. These samples provide a very interesting application to the classification process since they can all occur within local area each with their own physical properties which would affect navigation on top of these terrains.



(a) Woven pattern carpet

(b) Black thick carpet

(c) Soft gray carpet

Figure 1.3: Comparison of different types of carpet textures and materials



(a) Dirt

(b) Grass

(c) Pavement

Figure 1.4: Comparison of common outdoor terrains



(a) Snow

(b) Water saturated snow

(c) Ice

Figure 1.5: Comparison of cold weather related terrains

Chapter 2

System Design

In order to facilitate the collection of high resolution terrain samples, we require a robust robot and vision system to acquire images of terrains in a coordinated and consistent manner. To accomplish this task we use a small but rugged mobile robot with a robotic arm that is used for manipulating a machine vision camera around a terrain sample. In addition to the manipulation requirements, special attention must be made to the lens specification process to ensure a maximum field of view for a limited working distance. In this chapter we summarize the criteria of selection of the mobile robot, robotic arm, and camera with discussions on how each component meets the required constraints for the overall system.

2.1 Pioneer 3-AT Mobile Robot

The Pioneer 3-AT Mobile Robot serves as a platform for supporting equipment to a desired location for an image sample. Within its metal frame, three 12 volt lead acid batteries supply power to itself, the 7 degree of freedom Cyton Gamma 300 Arm, and a wireless networking adapter. A more detailed overview of the power distribution network can be seen in Figure 2.2. The robot drives its all-terrain wheels through a dual internal tread network each connected to two motors accompanied by high-resolution shaft encoders for tracking distance traveled. The robot uses skid-steering which means the wheels on a particular side operate synchronously together but independent from the other side. Full specifications for the robot can be found in Table 2.1.

The robot is manipulated over a wireless network from a remote laptop running control software. The commands are then transported through a RS232 serial bus to the on-board microcontroller running ARCOS firmware. Various commands can be sent

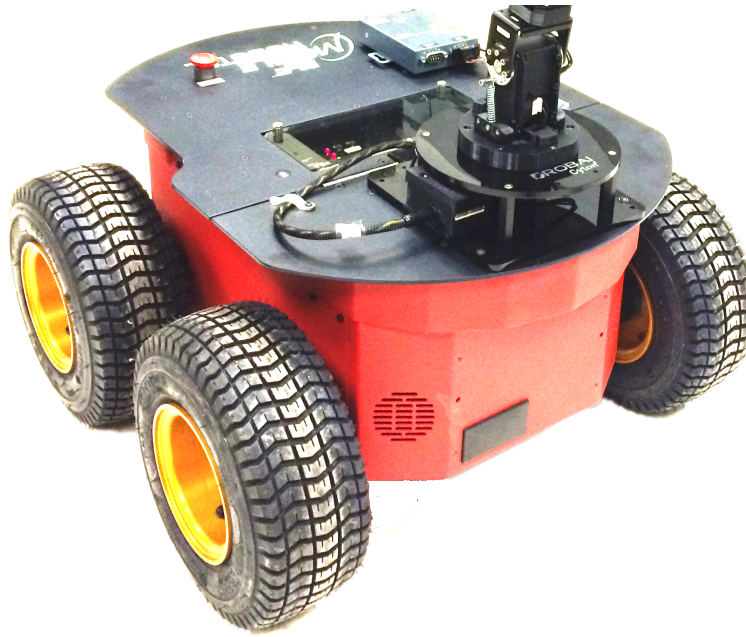


Figure 2.1: Pioneer 3-AT Mobile Robot

to the robot for teleoperation such as which direction to move and to stop. In return, the robot provides data on how fast the robot is moving, the current mode of operation, and an estimation of how far the robot has moved away from its origin location.

2.2 Cyton Gamma 300 Robot Arm

The robotic arm manipulator (Figure 2.4 attached to the front top of the Pioneer Mobile Robot), is a key component to the collection of ground terrain samples. The arm has seven servo motors defined as:

- **Shoulder Roll:** This is the lower most servo and controls the arms rotation with respect to the base, this allows the arm to move 300° in front of the robot.
- **Shoulder Pitch:** Right above the roll servo, the pitch servo is used for lowering and rising the arm, this manipulation has the greatest effect on the final height of the arm.
- **Elbow Roll:** Half way up the arm, this roll servo allows the arm to begin manipulating into complex forms.

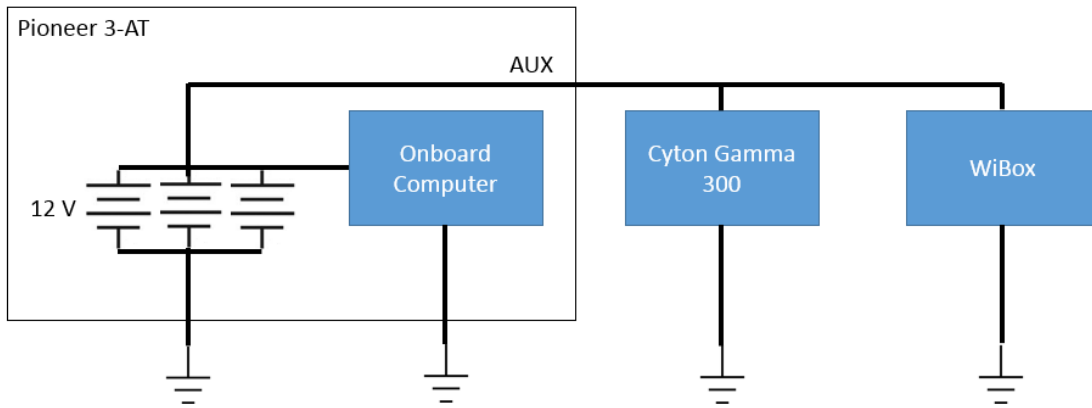


Figure 2.2: Mobile robot power distribution diagram

- Elbow Pitch: After the roll servo, the pitch can be adjusted to moved inwards or outwards.
- Wrist Pitch: this servo controls the pitch orientation of the gripper.
- Wrist Roll: The final orientation of the gripper is set by this servo.
- Wrist Yaw: This servo controls the gripper and causes the gripper to compress or contract.

The combination of each servo angle forms a single kinematic equation for finding X,Y,Z coordinates of the gripper head. The included software and API can also solve for the inverse kinematics of the system. That is, given an XYZ of the gripper location what are the necessary servo angles for reaching this position. Figure 2.3 shows how power is distributed to the arm.

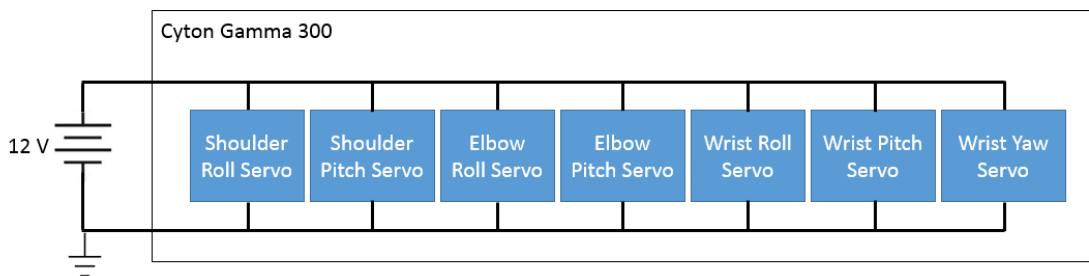


Figure 2.3: Cyton Gamma 300 servo power distribution

Attached to the gripper is a Basler Ace Area Scan camera, details of the camera can be found in Section 2.3. The grippers orientation is locked to always maintain its direction to a point on the ground six inches in front of the robot. This requires a coordinate frame transformation as described in Section 3.3. Full specifications of the arm can be found in Table 2.2.



Figure 2.4: Cyton Gamma 300 Robot Arm

2.3 Basler Ace Area Scan Camera

The camera of choice for this work is the Basler Ace *acA2040-90uc*; this camera is suited well due to its high resolution and global shutter. Basler also has a comprehensive API

for interfacing with the camera to easily adjust features such as exposure and white balance. Full specifications can be found in Table 2.3.

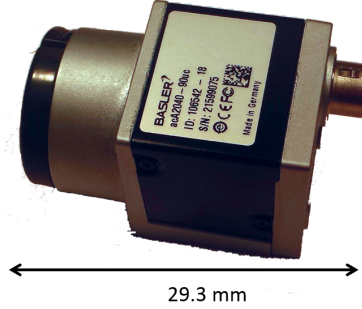


Figure 2.5: Basler Ace *acA2040-90uc*

2.3.1 Lens selection methodology

Choosing a lens for the Basler Camera required careful considerations to the intended use. The lens must meet a few criteria as limited by the design of the entire robot system:

- The maximum reach of the arm which affects the maximum angle that can be reached
- The arm can only support up to a 300 g payload
- The arm distance to the sample area

Based on the criteria given, a single focus lens was better suited for this design. The was due to the nature of lenses that support multiple focal points containing additional lenses which would bring the payload size to greater than 300g. To determine the appropriate focal length, we used the following equation,

$$\frac{1}{f} = \frac{1}{d_o} + \frac{1}{d_i}, \quad (2.1)$$

where f is the focal length, d_o is the distance of the object to the camera sensor, and d_i is the distance of the lens to the object. From this proportionality, we determined that the smallest available focal length would provide us with the greatest working area.

We chose a focal length of 25 mm and its specifications are described in more detail in Table 2.4.



Figure 2.6: TechSpec 25mm Camera Lens

2.4 Software and Central Processing

To accommodate the communication and processing needs of each component, a laptop is used to interconnect all of the devices. The laptop needs to run separate programs for controlling each component of the robot. The robots movement is controlled by a teleoperation program for sending velocity commands to the motors. In order to connect to the robot, a mobile cellphone based WIFI hotspot is used to host the robot and the laptop on the same network. The arm connects to the laptop over USB2.0 and has a graphical user interface for manipulating the arm. Finally, the camera is controlled through a custom script that is ran through the arm software API. This script is provided in Appendix B.

Table 2.1: Specifications of the Pioneer 3-AT Mobile Robot [2]

Construction	Body: 1.6 mm aluminum (powder-coated) Tires: Reinforced Pneumatic
Operation	Robot Weight: 12 kg Operating Payload: Tile/floor – 12 kg Grass/dirt – 10 kg Asphalt – 5 kg
Skid Steering Drive	Turn Radius: 0 cm Swing Radius: 34 cm Max. Forward/Backward Speed: 0.7 m/s Rotation Speed: 140°/s Max. Traversable Step: 10 cm Max. Traversable Gap: 15 cm Max. Traversable Grade: 35% Traversable Terrain: Asphalt, flooring, sand, and dirt.
Batteries	Run Time: 2-4 hours w/3 batteries Supports up to 3 at a time Capacity: 7.2 Ah (each) Chemistry: lead acid Hot-swappable Batteries: Yes
User Control Panel	MIDI programmable piezo buzzer Main power indicator Battery charge indicator 2 AUX power switches System reset Motor enable push button

Table 2.2: Specifications of the Cyton Gamma 300 Arm [77]

Axis Range	Total 7 independent axes Shoulder Roll (Spin): 300° Shoulder Pitch (Articulate): 210° Elbow Roll (Spin): 300° Elbow Pitch (Articulate): 210° Wrist Yaw (Articulate): 210° Wrist Pitch (Articulate): 210° Wrist Roll (Spin): 300°
Mechanical	Total weight: 1.2Kg Payload at full reach: 300g Payload at mid reach: 350g Arm length : 53.4cm (base to tip) Reach: 48cm Maximum linear arm speed: 20 cm/sec Maximum speed (free move): 100 cm/sec Maximum joint speed 30 rpm Repeatability +/-0.5mm Gripper: 2 parallel fingers, Maximum opening: 3.5cm
Electrical	Battery supply: 12V DC 2A

Table 2.3: Specifications of the Basler Ace Area Scan Camera [8]

Resolution	2048 px x 2048 px
Pixel Size horizontal/vertical	5.5 μ m x 5.5 μ m
Frame Rate	90 fps
Interface	USB 3.0
Video Output Format	Mono 8, Bayer BG 8, Bayer BG 12, Bayer BG 12p
Pixel Bit Depth	10, 12 bits
Lens Mount	C-mount
Digital Input	1
Digital Output	1
General Purpose I/O	2
Power Consumption (typical)	2.6 W
Weight	80 g
Sensor Technology	Progressive Scan CMOS, global shutter
Max. Image Circle	1 inch
Sensor Size	11.26 mm x 11.26 mm

Table 2.4: Specifications of the TechSpec High Resolution Lens

Focal Length FL (mm)	25
Maximum Camera Sensor Format	1"
Field of View, 1" Sensor ($^{\circ}$)	104 mm – 35 $^{\circ}$
Working Distance (mm)	150 – ∞
Aperture (f/#)	f/1.8 – f/22
Numerical Aperture NA, Object Side	0.031
Distortion (%)	< -2.0
Number of Elements (Groups)	7 (6)
Coating	MgF_2
Coating Specification	$\frac{1}{4}\lambda MgF_2$ @ 550nm
Outer Diameter (mm)	37.5
Length (mm)	45.3
Mount	C-Mount
Weight (g)	113

Chapter 3

Computer Vision and Robot Integration

As described in the previous chapter, the complete robot consists of:

- Pioneer 3-AT Mobile Robot
- Cyton Gamma 300 7 DoF Robotic Arm
- Basler Ace *acA2040-90uc*

The complete design as shown in Figure 3.1 was strategically put together in order to meet the objectives of this research. As a result, the final robot design and integration came to be through many design iterations and careful planning. This chapter describes the design considerations implemented for putting the pieces together, the communication networks and buses required for all the pieces to work together, and finally a robot coordinate frame designed for taking sample images with respect to the world.

3.1 Design Considerations

Below we list and describe the limitations and criteria for assembling and integrating the mobile robot platform.

- Arm mount - We mount the arm on the top of the mobile robot. This required an additional steel mounting plate to support the weight of the arm when facing in front of the robot.
- Payload weight - The arm can only support a maximum of 300 g, this is also not guaranteed for every possible position. To account for this, the camera and lens were chosen such that they only weigh 203 g.



Figure 3.1: Complete robot system design

- Maximum arm reach - In order to increase the size of the sample area, we iteratively adjusted the working distance to allow for a large sample area that could fit 33 sample points.
- Power - With the three 7.2 Ah batteries, the robot can run continuously for 2.5 - 3 hours. The laptop controlling the robot and additional components only has enough battery power to last 2 hours so this was the limiting factor to sample collection
- Communication - While the robot can communicate over WIFI, the camera and arm required a wired connection. This made it necessary to use an additional push cart to keep the laptop close to the robot. Wire management was also

important to make sure that the arm would not get tangled with the camera wire. To account for this, the camera cable was held over the arm during sample collection process.

3.2 Communication

All the components of the robot communicate through a multitude of buses for transporting and receiving data. In order to coordinate the data transmission, a laptop computer is used as a central host. Figure 3.2 shows the communication channels employed in this design. One can see that this system utilizes the following communication protocols:

- RS232 Serial
- USB 3.0
- USB 2.0
- IEEE 802.11 WIFI

3.3 System Coordinate Frame

By default the Cyton Gamma 300 arm moves with respect to its own internal coordinate frame set at the center of the manipulator base. Let this point be known as the origin or $O = (0, 0, 0)$, u_x represents a unit vector in the x direction, u_y represents a unit vector in the y direction, and u_z represents a unit vector in the z direction. Using this notation, a 3D point $p_a = (x_a, y_a, z_a)$ can be found for this coordinate frame

$$p = O + x_a u_x + y_a u_y + z_a u_z. \quad (3.1)$$

However, the base of the arm is not typically the camera's view. Instead the coordinate frame must be translated such that the center of interest or the new origin becomes the location of terrain sampling. With the coordinate frame centered at the

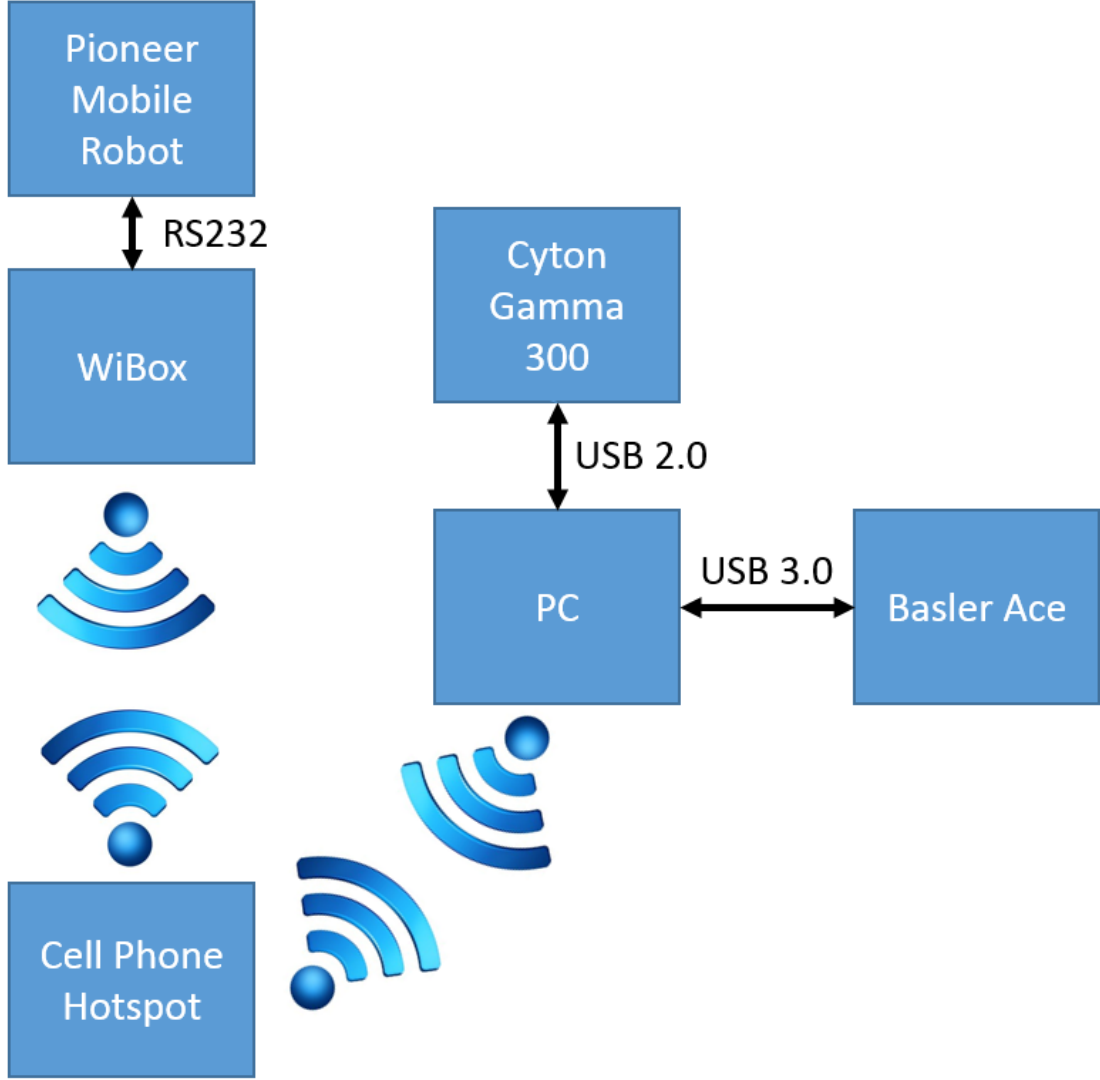


Figure 3.2: Mobile robot network design

surface point of interest, a uniform sampling across a hemisphere can be expressed with respect to the sample area. This translation can be represented as

$$p_b = (x_b, y_b, z_b) = \begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \\ 1 \end{bmatrix} = \begin{bmatrix} x_a + \Delta x \\ y_a + \Delta y \\ z_a + \Delta z \\ 1 \end{bmatrix} \quad (3.2)$$

where Δx represents the coordinate frame translation in the x direction, Δy represents the coordinate frame translation in the y direction, and Δz represents the coordinate

frame translation in the z direction. For this robot it was measured that the sample area is located 0 mm in the x direction, positive 190.5 mm in the y direction, and negative 279.4 mm in the z direction. This places the sample area positive 127 mm along the y coordinate frame of the robot and level with the ground as shown in Figure 3.3. These measurements account for the arm placement that is offset from the edge of the Pioneer robot. The new sample coordinate frame does not require a rotation since the original arm coordinate frame matched our needs.

3.3.1 Maximum sample area

In Section 2.3.1 the criteria for selecting a camera lens was carefully laid out as a function of characteristics designed to work with the overall system design. This sets the stage ultimately for mathematically finding the largest possible sample area given the constraints of the system. For this work, the sample area is defined as a square bounding area that is centered around a fixed point that is found in Section 3.3.

Figure 3.4 shows the maximum reach the arm is capable of as enclosed in red. Beyond the red area, the arm cannot extend any further and therefore limits the overall sample area.

The sample area $FOV = X^2$ where X is the length of one side of the sample area, can be found as a function of focal length f of the camera lens, the working distance W_d to the sample area, and the size of the CCD sensor S_{CCD}

$$FOV = \frac{S_{CCD} \times W_d}{f}. \quad (3.3)$$

Figure 3.5 shows this relationship between the camera and sample area. S_{CCD} is known from Table 2.3, W_d was measured to be 304.8 mm, and the focal length f was determined from the lens selection criteria in Section 2.3.1. This gives a FOV of 127x127 mm.

3.3.2 Smallest feature size

Now that we know the field of view, we can then calculate the smallest feature size that the camera can capture. Using the following equation,

$$SmallestFeature = \frac{2 \times FOV}{ImageResolution} \quad (3.4)$$

we discover that the smallest feature size is 60.8 μm per pixel. This information is useful for representing the amount of terrain data contained per image.

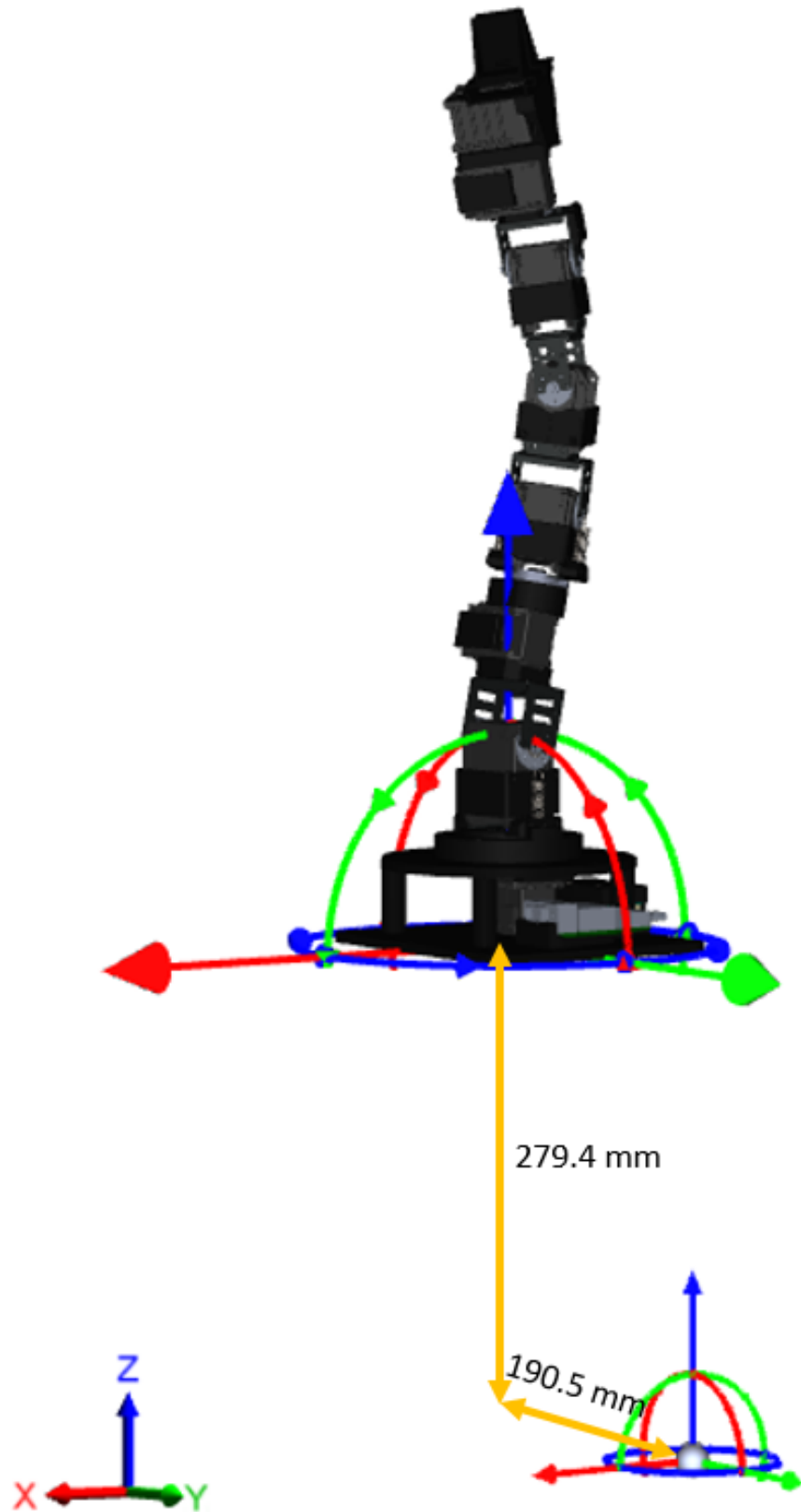


Figure 3.3: Coordinate frame transformation from arm coordinate frame to sample area coordinate frame. The sample area origin is denoted as a small sphere.

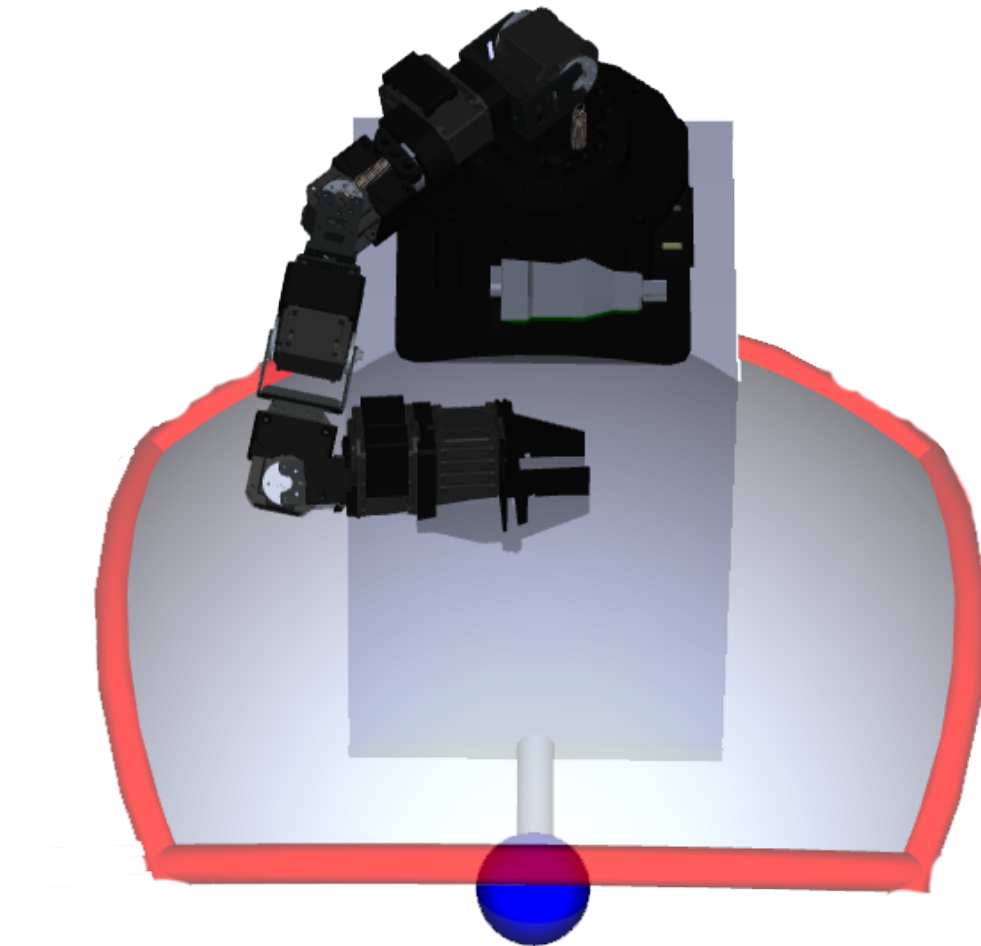


Figure 3.4: Maximum arm reach for a radius of 304.8 mm from sample area. Red borders show the furthest points the robotic arm can reach forming a partial hemisphere of coverage over a sample.

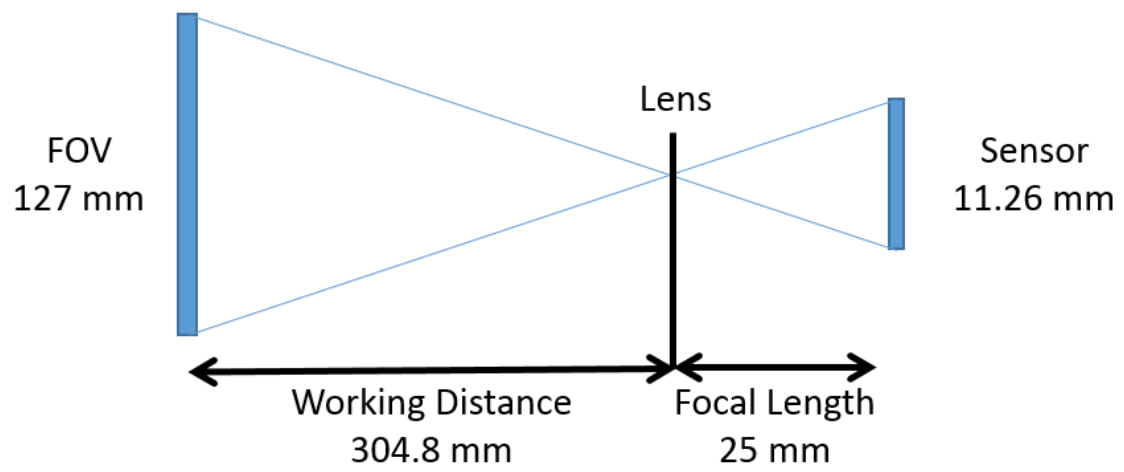


Figure 3.5: Field of view as a result of camera working distance

Chapter 4

Ground Terrain Database

In this chapter we describe the pipeline for collecting a sample. We start by determining a series of point locations for image capture, at each location a pose must be set to orient the camera with the sample, and finally a manipulation action sequence is designed to bring the robotic arm to each point successively and collect images. Figure 4.1 describes this process. We also examine the efficiency of the procedure and the storage requirements for collecting 50 samples.

The sample data set includes a multitude of terrains located in common outdoor and indoor environments including: carpet, tile, street paint, brick, concrete, gravel, dirt, mulch, pavement, metal, grass, ice, slush, snow, rocks, rubber mat, sand, and wood. Some of the outdoor surfaces are also collected in two forms of environmental conditions including wet and dry: concrete, pavement, mulch, and metal. A complete list of these samples can be found in Appendix D.

4.1 Capture Point locations

The location of each capture point was determined from the limitations of the Cyton Gamma robot arm ability to reach around a sample as discussed in Section 3.3.1. It was determined that there should be an arch of points along a center axis spanning from 45° to -45° in the x axis. On the arc 7 evenly spaced points were chosen each separated by 15° these points form the basis of the image sample collection and are referred to as *point1_1* to *point7_1* from positive x to negative x respectively. Furthermore, 4 additional corner locations were chosen to provide angles which would encounter a greater difference of environmental lighting effects from the arch. These points are referred to as *point8_1* which is located 30° up and 20° back from *point6_1*, *point9_1*

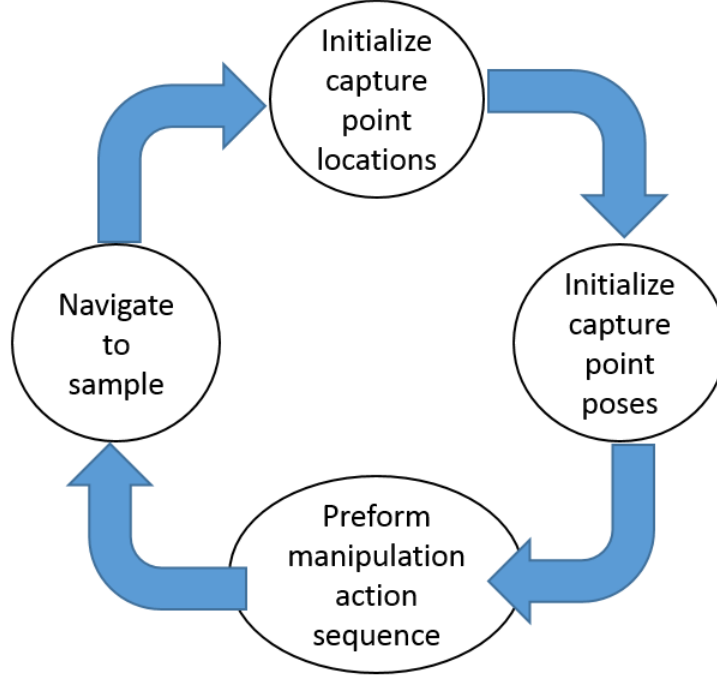


Figure 4.1: The process of taking samples follows this pipeline, positions are loaded, poses are set, and finally a manipulation action sequence occurs. This repeats for every sample.

which is located in located 30° up and 20° forward from *point6_1*, as well as *point10_1* and *point11_1* which are on opposite corners. Figure 4.2 denotes these points in yellow. We define these positions as fundamental measurement points.

4.1.1 Small angular changes for gradient computation

At every fundamental measurement (r, θ, ϕ) point, we have also measured two different angles $(r, \theta + \delta, \phi)$ and $(r, \theta - \delta, \phi)$ with $\delta = 5^\circ$ to examine the properties of small angular changes on illumination differences in samples. The first set of angles is denoted by purple and are post fixed with *_2* Figure 4.2. The second set of angles is denoted in green and are post fixed with *_3* in Figure 4.2. In total there are 33 collection points.

4.2 Camera Pose

For each of the 33 points, an arm pose is selected such that the arm is able to manipulate to the desired position and orient the camera towards a sample. Figure 4.3 demonstrates

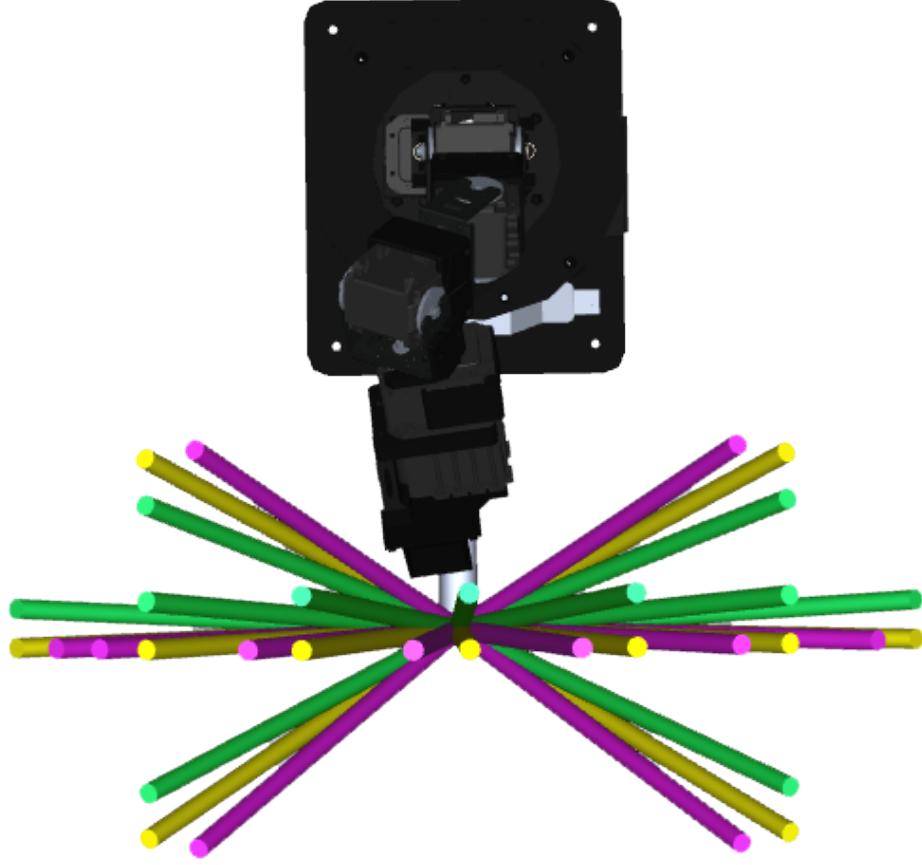


Figure 4.2: Top down view of capture point locations

one such pose, here the arm is oriented in the negative x direction with a 5° pitch to prevent the arm from hitting a joint movement limit. The gripper is parallel to the sample at *point4_1* since it is located directly above the sample. A complete listing of all points and poses can be found in Appendix A.

4.3 Manipulation Action Sequence

After determine the positions and poses, a manipulation action sequence was made to drive the robot arm and camera to each position. An action sequence is defined as the



Figure 4.3: Arm placed in position *point4_1*

paths the robot arm takes to settle into each fixed position. The arm moves to each point starting at *point1_1* and then proceeds to *point1_2* and *point1_3* it continues this until it reaches *point11_3*. At each point the software is paused to allow for an image to be taken. The arm is not intelligent to form paths that would prevent it from colliding with itself or reaching joint movement maximums. As a result, a path was determined for certain points through a combination of waypoints that is, in order to reach a point the arm must manipulate to one or more additional locations first before it can get to the targeted point. One such example of this is shown in Figure 4.4, here the arm is unable to reach *point3_1* since the manipulation from *point2_3* will cause a collision on itself as shown in Figure 4.5 Therefore by extending the arm away from the targeted point and instructing it to come back from the extension, the arm is now able to reach

the desired location without issue.

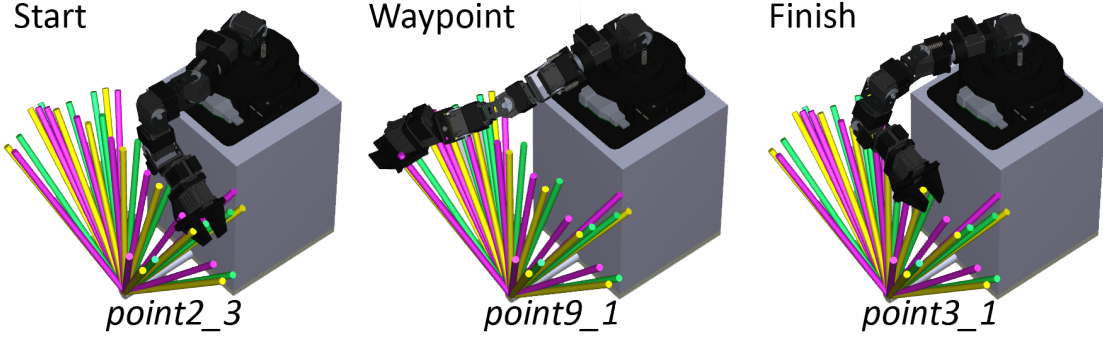


Figure 4.4: Sequence of arm movement from *point2_3* to *point3_1* by waypoint *point9_1*

4.4 Collection of Samples

Now that we have established a consistent and repeatable method for collecting samples, it is best to address the process for manipulating the robot and collection samples out in the wild. Figure 4.6 shows the mobile robot at a sample location, the arm is standing by waiting to be given the signal to collect images, a laptop on a pushcart acts as a central command center for controlling the robot, arm, and camera.

4.4.1 Camera gain, white balance, and exposure

Before collecting a sample, a simple initialization routine is performed to find an optimal gain, white balance, and exposure setting. Algorithm 1 gives a general outline for the process. First the arm is moved to *point4_1* which is located directly above the sample, then the camera grabs a series of images and readjusts the parameters until they meet a threshold that the camera software determines to be desirable. Once the value is found, the camera stores this value for the entire duration of a single sample taking period.

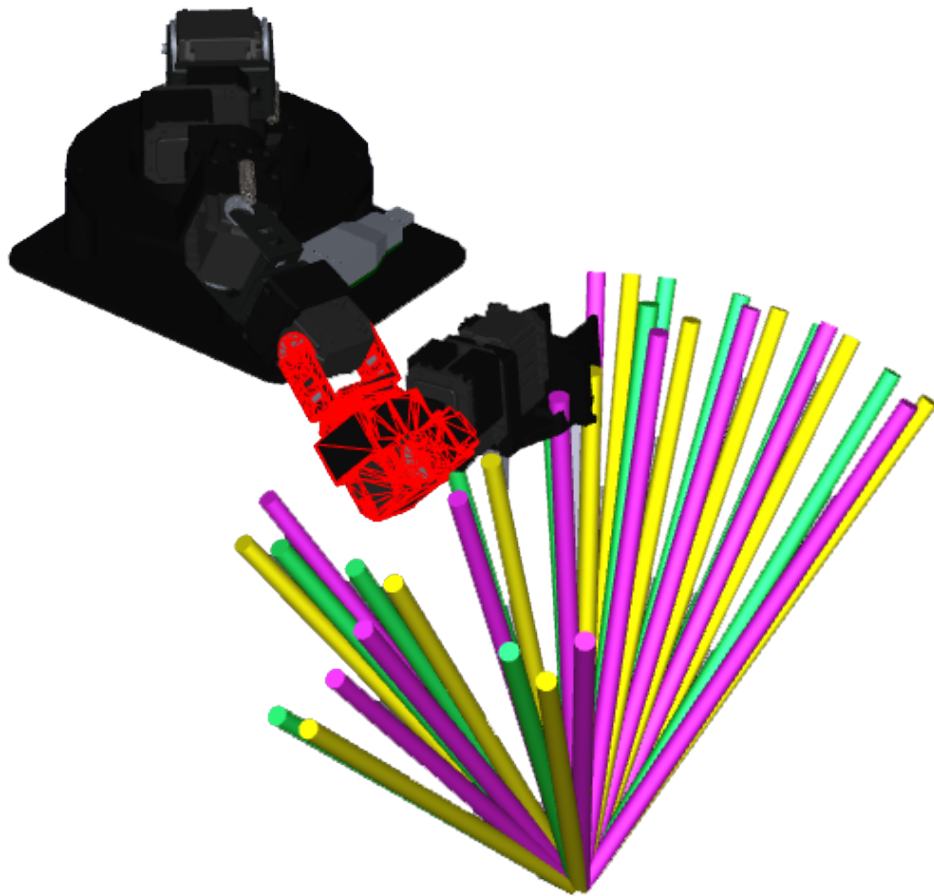


Figure 4.5: Example of arm movement failure moving from *point2_3* to *point3_1*

We keep a record of these values in a text file that accompanies each sample.

```

while (gain/whitebalance/exposure)! = AutoCalc_Off do
    Grab a new image;
    ++n;
    if  $n > 100$  then
        | throw error ("Unable to find suitable value" );
    end
end

```

Algorithm 1: Process for setting gain, white balance, and exposure values



Figure 4.6: Robot setup to capture a ground terrain sample outdoors

4.4.2 Digital storage requirements

Images are captured as an 8 bit bitmap, this means that for each pixel there are 8 bits or 1 byte of color data for the values of red, green, and blue. As a result images are always non compressed and contain exactly the same amount of data per photo. Each photo is 2040x2046 pixels, therefore by using this equation:

$$ImageSize(MB) = \frac{3 \cdot W \cdot H}{10^6} \quad (4.1)$$

we can solve for a single image storage requirement. Therefore, each image uncompressed requires **12.52 MB** of digital storage space.

One can see how quickly the storage requirements grows with each sample of 33 images,

$$SampleSize(MB) = ImageSize \cdot I \quad (4.2)$$

where I is the number of images per sample. This comes out to **413.16 MB** per

complete sample. Therefore the entire ground terrain database size is,

$$DatabaseSize(MB) = SampleSize \cdot N \quad (4.3)$$

where N is the number of samples in the database. This comes out to **20,658 MB** or **20.66 GB** required to store the entire database uncompressed. These 50 samples contain 1,650 individual images in total.

Chapter 5

Classifying Ground Terrain Surfaces

In this chapter, we discuss in detail both Texton Boost and Angular Gradient Differential Histogram Classification for the purpose of classifying a multitude of ground terrain surfaces. Both methods follow a similar means for initial training but diverge in classification and recognition. Both methods rely on k-means based texton dictionaries and dense per-pixel texton mappings of images. The general summary for this method is described below and in more detail in the following sections:

Training

1. Initial K-means training
 - (a) Take a selection of 4 images from each sample providing 200 total sample images
 - (b) Run the images through a 19×19 filter bank containing
 - (c) Create a collection of subsampled pixels selecting 1,000 random pixels per filtered image
 - (d) Perform K-means clustering to locate 100 centroids
2. Texton histogram training
 - (a) Calculate a dense Euclidean distance to each centroid per each training image
 - (b) Sort distances in ascending order
 - (c) Weight the top eight closest distances
 - (d) Create a histogram based on softbins for each training image

Classification

1. Filter all the images
2. Create texton histograms for every image in the database
3. Find the distance between each histogram from the training data
4. Sort the distances, the closest distance is its label

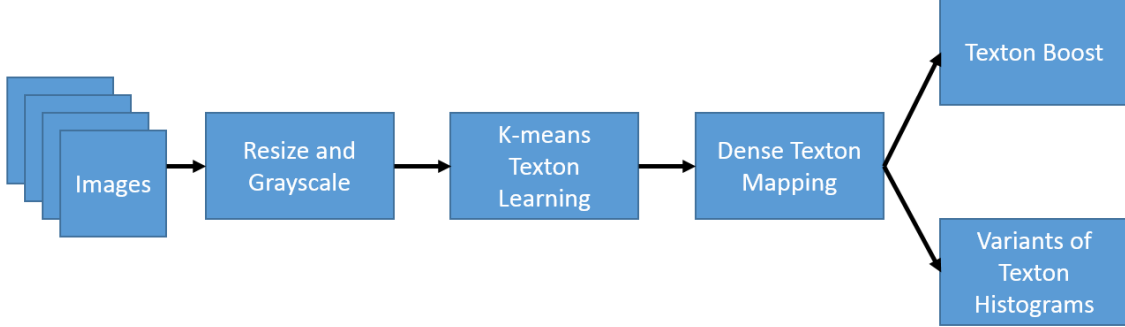


Figure 5.1: High level overview of terrain classification process

5.1 Image Preprocessing

In order to process these images in any reasonable amount of time, each image is preprocessed in the following ways. first, each image is re-sized using downsampling and anti-aliasing to $\frac{1}{4}$ of the original image size (500×500 pixels). Second, the images are averaged and converted to grayscale. Neglecting the RGB color channels for these images is an important criteria in this process since the targeted area of interest lies with changes in illumination between viewing angles which does not require color. These steps greatly reduce individual file size to become a more reasonable **245 KB**.

5.2 K-means based Texton Learning

We use k-means to create 100 clusters or textons that form the basis of a texton dictionary to be used for mapping later on. In order to map each pixel to a representative texton label, we first filter the images against a 19×19 convolutional filter bank to extract edge features and gradients from within the images. We then subsample 1,000 pixels from each filter response to create our training set for k-means. Finally a texton

dictionary is created by running k-means clustering on the collection of all of the sub-sampled pixels. Figure 5.2 shows the process of converting from one training image to the texton dictionary.

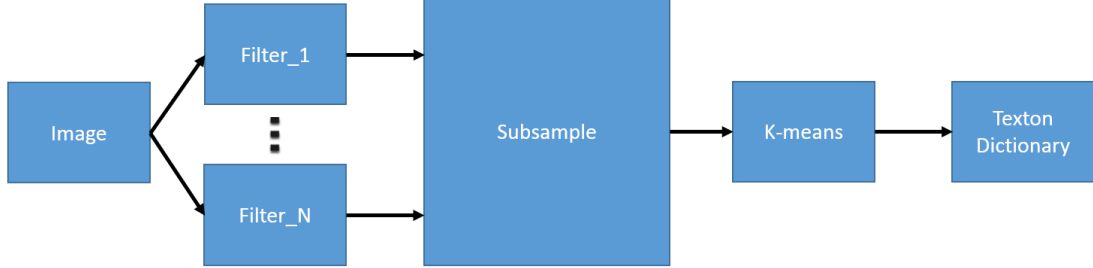


Figure 5.2: Process for converting training images to texton dictionary

5.3 Dense Texton Mapping

The second part of training involves running the same 200 training images through a texton mapping scheme. Using the same filter bank responses, we calculate the Euclidean distance of each pixel against each of the 100 textons using the following equation,

$$d_i(pixel) = (R(pixel) - t_i)^T (R(pixel) - t_i), \quad (5.1)$$

where $d_i(pixel)$ is the distance between the pixel response and the i th texton; $R(pixel)$ is the 15×1 response vector at the current pixel, t_i is the 15×1 response vector associated with texton t_i . The collection of these calculation is a vector with 100 distances for each pixel. Figure 5.4 shows a collection of dense texton maps, here one can see how edges and grooves in terrains tend to classify similarly. The results of terrains with depth based features provide the most interesting responses, grass, gravel, and carpet can clearly be seen through the texton mapping while dry concrete and pavement are difficult to distinguish.

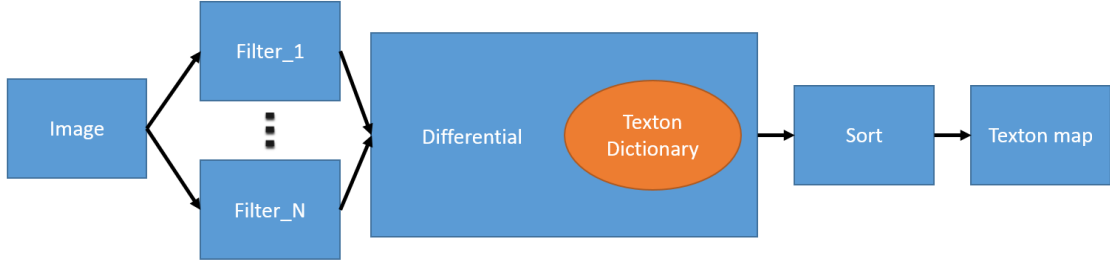


Figure 5.3: Process for converting training images to dense texton maps

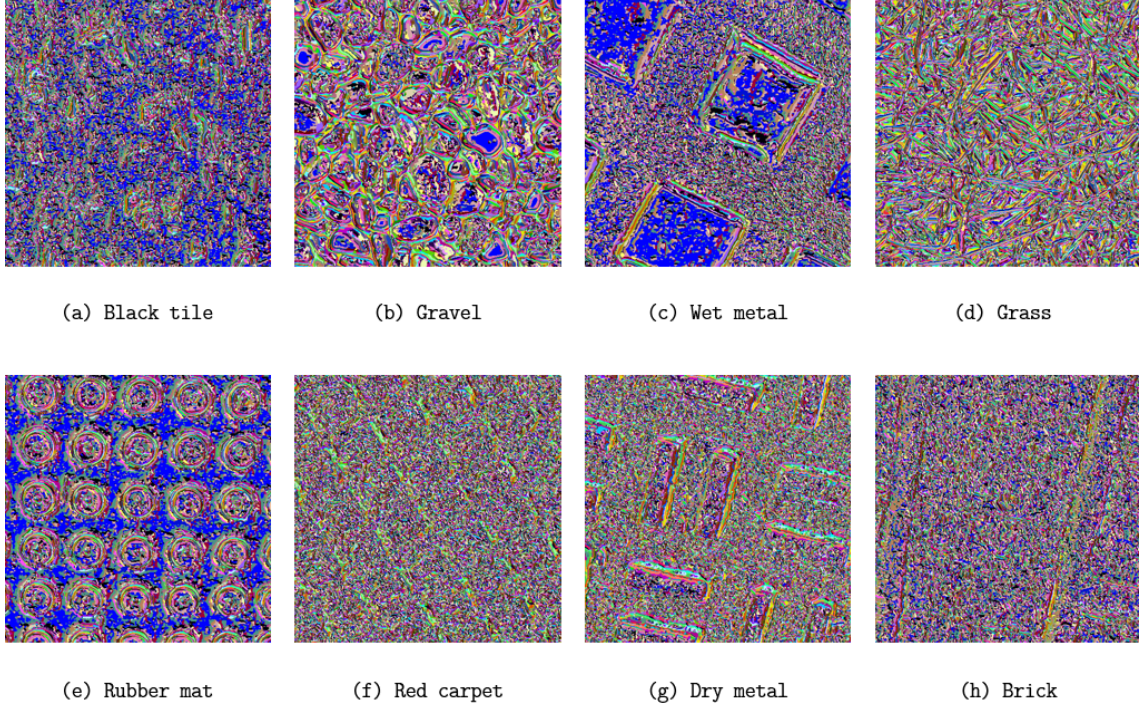


Figure 5.4: Select results from the texton map training

5.4 Texton Boost

This method iteratively builds a strong classifier as a sum of weak classifiers, to select discriminative features. The set of weak classifiers forms a decision stump that is then thresholded to provide a threshold feature response [85, 91]. The response is then put into a set of shared feature classes to form a single feature that can classify multiple classes at the same time. This becomes a learned strong classifier that comes from the addition of weak classifiers.

5.5 Weighted Histograms

We used a method of weighted histograms over traditional histograms to account for close similarities between textons. If this was done with a regular histogram we would just take a texton map and count each cluster occurrence and place them into 100 bins each representing a single texton. Instead, we weight the eight closest distances using the following equation,

$$softbin(i) = 1 - \frac{d(i)}{\sum_{i=1}^8 d(i)}. \quad (5.2)$$

Each bin in the histogram is now the sum of weighted occurrences. We call this a weighted softbin histogram. Figure 5.5 shows a selection of softbin weighted histograms.

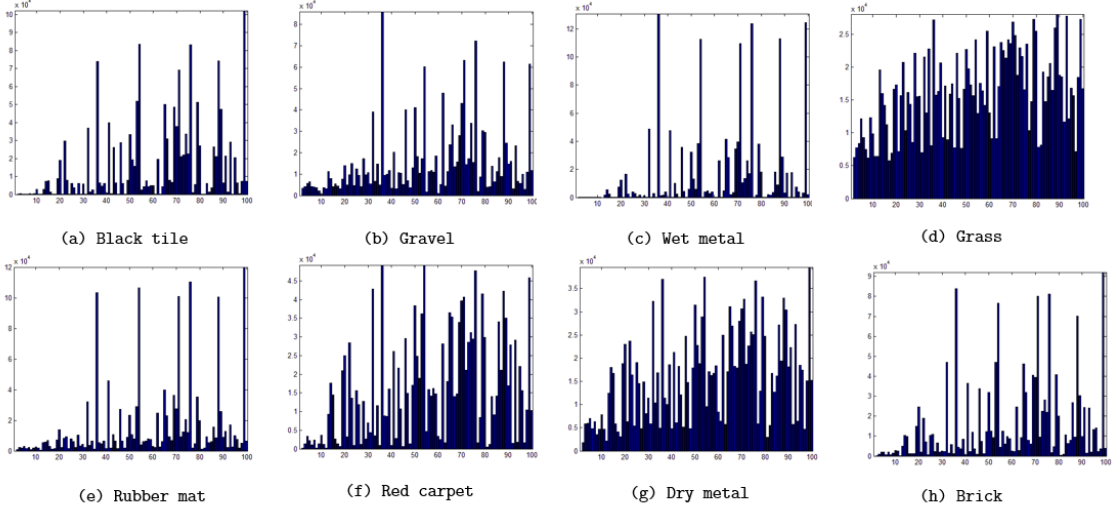


Figure 5.5: Weighted histograms for multiple ground terrains

We perform this step across every image both for training and testing. The collection of training image softbin histograms form the working data set for classification.

5.6 Classification

Now that we have trained the system through k-means, texton mapping, and weighted softbin histograms, we can begin to classify outdoor surfaces. Experiments were determined to best utilize the trained data. Here we pick three such methods for classification. Each is a trade off between robustness of classification and additional sample

requirements. We also utilize multiple histogram distance metrics to fine tune the classification.

5.6.1 Single Angle Histograms

As a preliminary course of action, we examined the success of correctly classifying terrains through single angle histograms. The texton histogram for each angle was compared against a training set consisting of all the texton histograms from all of the images collected. This provides a benchmark for the remaining experiments. To prevent training/testing overlap, we have split each image in half to reserve the right side images for training and left side images for testing.

5.6.2 Eleven Angular Gradient Histograms

From the 33 images taken per sample, 11 of which are considered to be fundamental points. Using these points, we adjust the terrain dictionary to now contain histograms as the combination of 11 angles through averaging. We then find the distance between all 11 histograms to 11 new histograms for testing. We expect that this method will have a very high classification rate due to the number of angles required but ultimately the goal of this work is to achieve a high recognition rate using as few angles as possible.

5.6.3 Pair-wise Angular Gradient Histograms

Following a similar method as with the eleven angular gradient histograms, instead we now take a pair of capture angles with a small delta between them. Using a single pair consisting of a fundamental angle and a delta angle we build one training histogram through different combination methods including; subtraction, averaging, and taking the bin minimum. After iterating over all images to build a terrain dictionary we then classify terrains through both Euclidean distance and histogram intersection.

Chapter 6

Results

In this chapter, we examine the performance of the methods outlined in Chapter 5. We explore the performance of Texton Boost and compare the results to Angular Gradient Differential Histograms. Lastly, we explore the precision and accuracy of these methods and discuss conditions for optimal system performance. Appendix D contains a list of samples and their respective numeric and written labels.

6.1 Angular Gradient Differential Histogram Classification

6.1.1 Eleven angle histograms

When classifying terrains from all eleven fundamental angles, we find the system to be able to correctly recognize all terrains at a rate of 98%. However, this is not a fair representation of the data due overlap in the training and testing data. Figure 6.1 shows a graphical representation of the confusion matrix. Here the axis represent the label number and the diagonal of the matrix represents instances of a training point correctly matching a testing class. The only material to not be correctly classified to its sample was terrain number 20 (grass) to sample 22 (grass3). In total there is only as many testing points as there are samples. To preform a more robust analysis of this method, we would need significantly more samples.

6.1.2 Single angle histograms

Lets now begin to challenge the system. By examining single angle histograms, we compare an image from all available angles against a training set that is comprised of the histograms of all possible angles. Here we see in Figure 6.2 an overall recognition

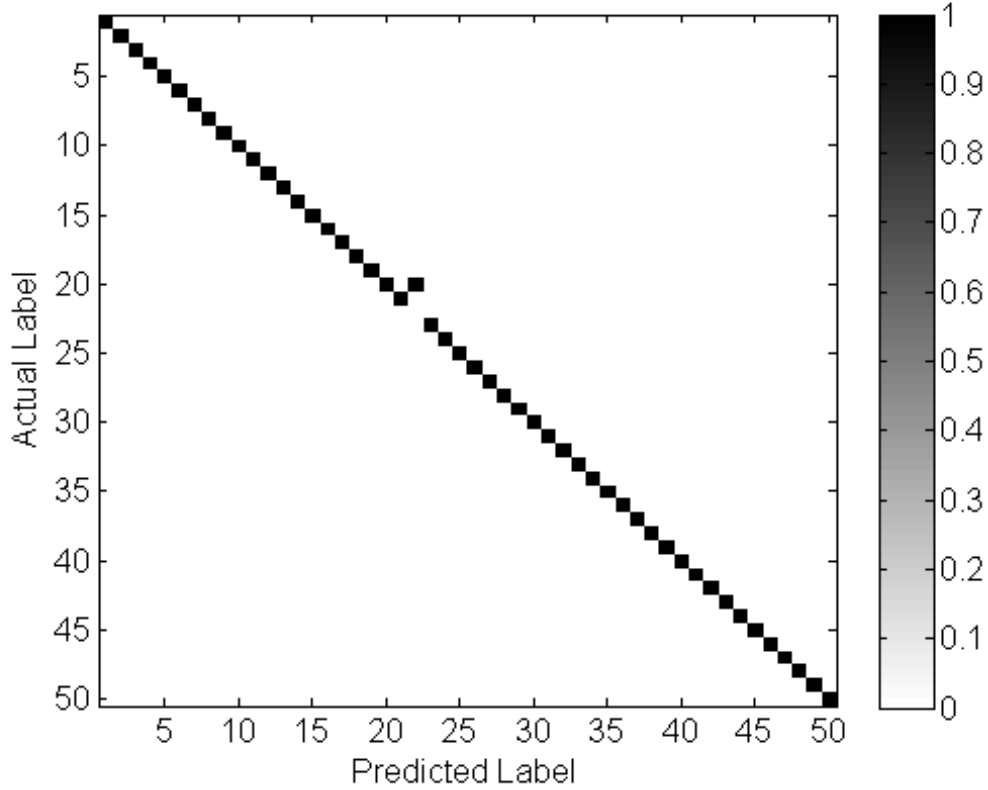


Figure 6.1: Confusion matrix for eleven angle histogram classification. Overall recognition rate is 98%

rate of 88.42% when the training set contains 1,650 histograms. The confusion for this method mostly stems from dry and wet terrains, with an overall good recognition across unique terrains.

6.1.3 Pairwise angle histograms

After examining both eleven and pairwise angular histogram methods, let's now examine the success of classifying terrains through a combination of angle pairs. For each experiment, we vary the method of histogram combination and classification metric. To combine pairs of histograms we analyze the performance of subtracting, averaging, and taking the minimum bin between the pair histograms. For classification we examine the case of using Euclidean distance and histogram intersection [12]. The histogram intersection provides a percentage of how closely related the sample under test is to a

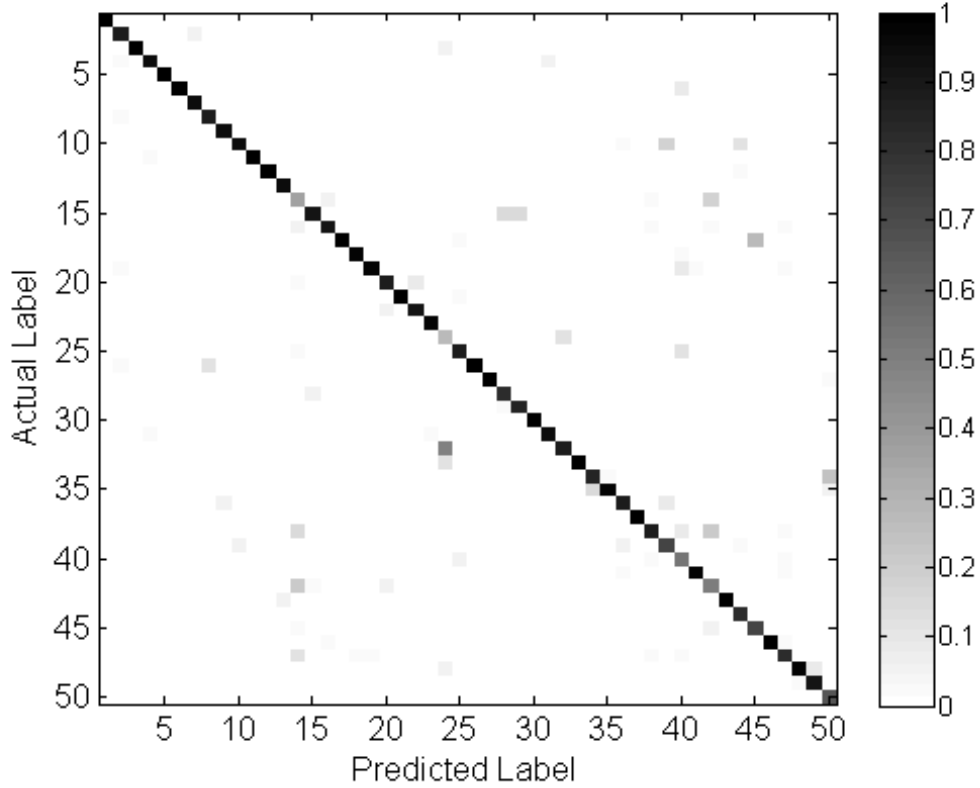


Figure 6.2: Confusion matrix for single angle histogram classification. Overall recognition rate is 88.42%

pair in the set of training pairs.

In Table 6.1, we examine the recognition rate across the three possible combinations of pairs of angles. That is, we examine the success of a fundamental angle paired with the first delta, a fundamental paired with its second delta, and the combination of both delta angles. We also compare the recognition rate of single angle histograms but with the same reduced training set which comes as a consequence of the pairwise training process. It is important to note that for single angle histograms, the recognition rate is slightly smaller (a reduction of 2%). This is due to a reduction in training points of $\frac{2}{3}$ or 1,100 less training points for comparison. We also note that variations between pair angles used for training slightly effect the recognition rate. As such, we find that pairing angles $_1$ with $_2$ provides the highest rate of recognition.

Next, we look at Table 6.2 to see which combinational method provides the best

Table 6.1: Recognition rates for different combinations of pairwise angle histogram classification (mean only)

_1	_1 and _2	_1 and _3	_2 and _3
86.18%	89.45%	87.45%	88.55%

recognition rate. Surprisingly we find that subtracting angle histograms provides a very poor result. After further review, we find that this is due to a loss of almost all terrain texture information embedded within the histograms. For example, an image of a terrain and an image of the same terrain but captured at a slightly adjusted angle, will contain almost exactly the same texture information where instead the only differences in composition will be as a result of differences in light reflection. Overall, across most terrains, the resulting histogram will contain too little distinguishing features to accurately classify these terrains correctly. However, Looking at the average bin values between terrain histograms provides the best results. Likewise, with the results of the subtraction example we find that the knowledge of the natural lights effects on viewing angle has an important effect to the overall recognition. By taking the average between histograms bins, we find that we are not changing the texture composition of a terrain, but rather, we are now incorporating information about natural light from these viewing angles which is therefore causing the training data to become less invariant due to the effects of natural illumination and provides a much better recognition. We also anticipate, that more training data would also further improve this result.

Table 6.2: Recognition rates for multiple methods of pairwise angle histogram classification

	Subtraction	Average	Min
Euclidean Distance	20.73%	86.91%	85.82%
Histogram Intersection	11.45%	89.45%	82.36%

Lastly, in Figure 6.3 we find that the confusion of this method is overall much better as compared to single angle histograms. We find that this comes from this methods ability to better classify both wet and dry terrains although still not perfectly.

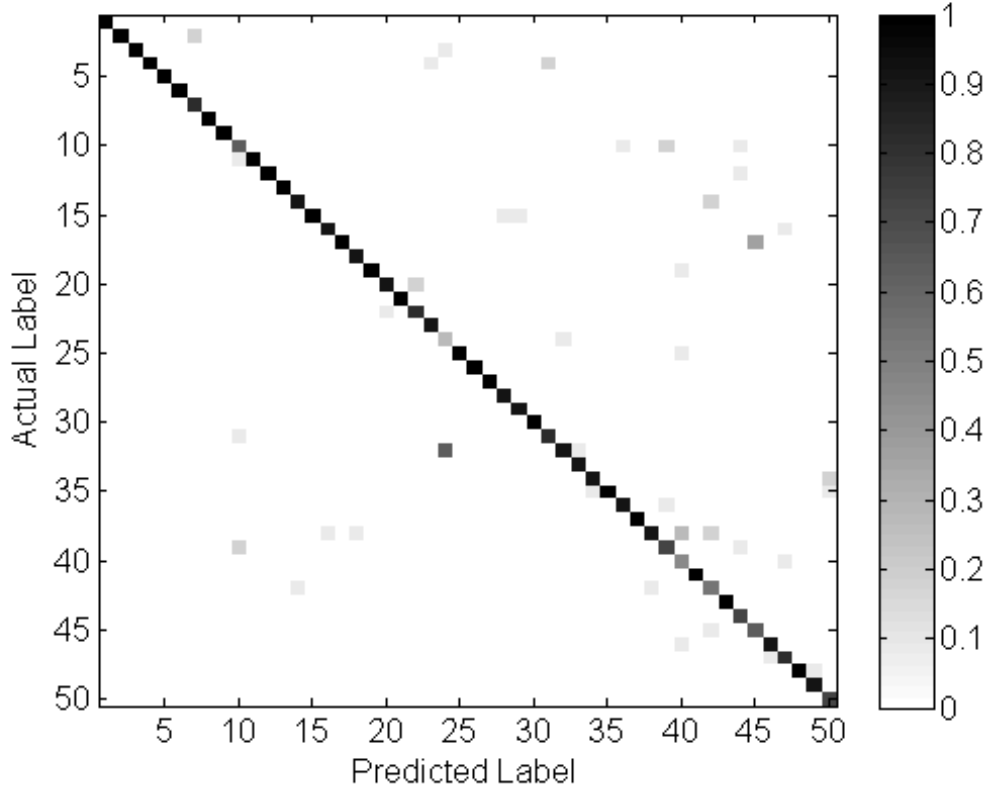


Figure 6.3: Confusion matrix for pairwise angle histogram classification. Overall recognition rate is 89.45%.

6.2 Texton Boost Performance

We also further compare our method to Texton Boost. Using the images of single angles, we show the results of this method in Figure 6.4. We find that overall, the algorithm is able to correctly classify a given terrain 83.68% of the time. In particular, the algorithm tends to confuse terrains such as wood with carpet and sand with dry concrete. Overall, this method does not work as well with our set of data. We find that this is mostly a factor of the composition of the data including very similar images of terrains only adjusted by viewing angle and direction. The algorithm attempts to create segments from these images however the homogenous nature of many of these samples makes it difficult to find significant areas of interest that would be used to further make correct classifications.

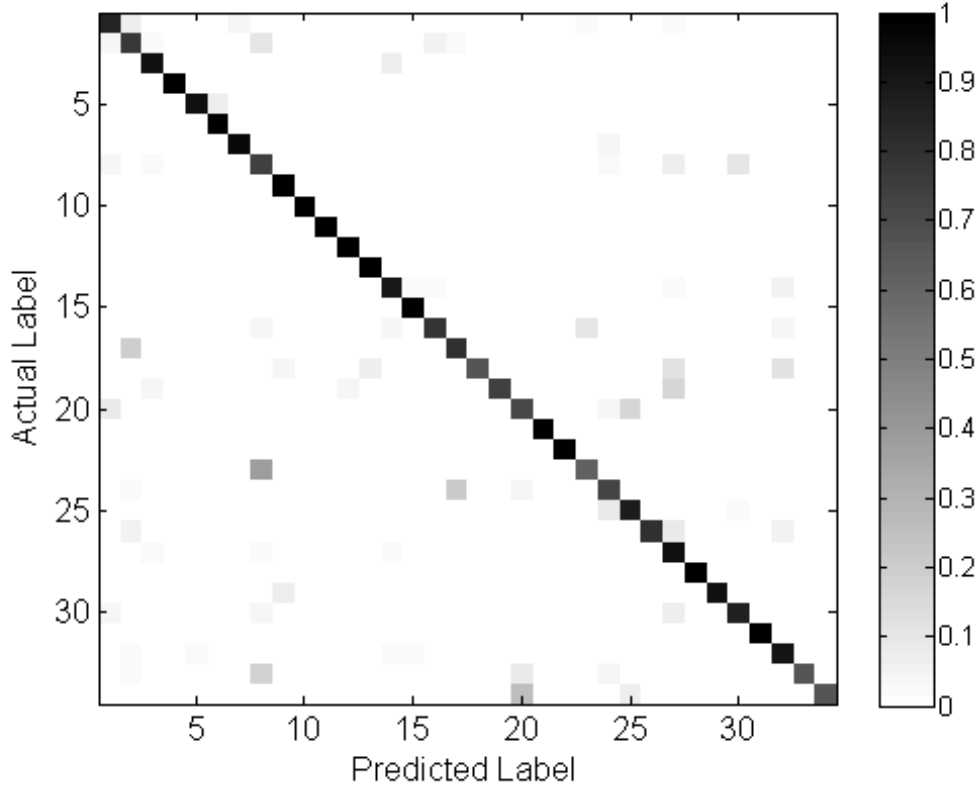


Figure 6.4: Confusion matrix of Texton Boost performance in image form, overall recognition rate of 34 unique terrains is 83.68%

6.3 Discussion

Overall, the recognition for both single and pairs of angles is very good. We find that that the recognition for single angles is much larger when a significantly larger training set is used. When constraining the single image classification to a smaller training set, we find that the pairwise angle classification performs significantly better. In Figure 6.5 we see that overall in the case of pairwise recognition we achieve 100% recognition across more samples than with single angle histograms.

Particular interest to this work is to instances for where we are unable to correctly classify a terrain. For example, ice, has an overall recognition rate of 27%. When looking at Figure 6.3, we can see that ice (label 24) is most often incorrectly recognized as slush or a mixture of ice and snow (label 32). Figure 6.6 investigates the similarities of these terrains by examining their texton maps, here we see that the two terrains

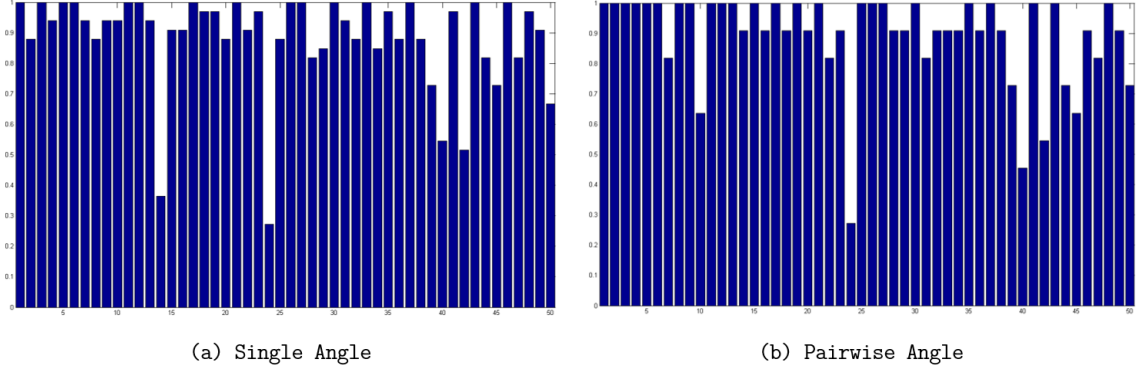


Figure 6.5: Comparison of 100% sample recognition between single angle and pairwise classification. Single angle has 15 instances of 100% recognition, pairwise has 24 instances of 100% recognition.

contain very few textons in general and many of them are the same. However, for this particular example and for the overall application target for this work, one would generally combine autonomous navigation profiles to act similarly for both snow and mixtures of snow and ice.

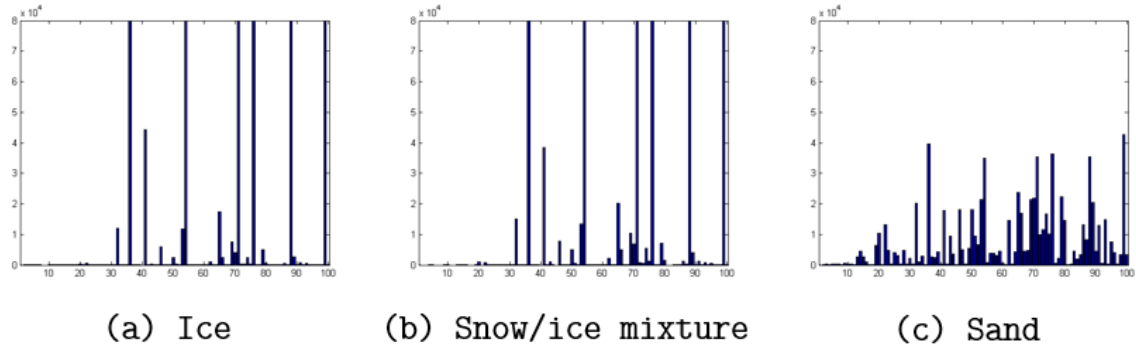


Figure 6.6: Texton histograms for ice and snow/ice mixture terrains (*histogram_intersection* = 95%). The histogram for sand is also added to provide further comparison to an additional sample.

Lastly, we also compare the recognition of wet surfaces against dry. We find that the overall recognition rate for wet/dry surfaces is 86.30%. Figure 6.8 shows histograms for a few of the wet and dry terrains, note how many of the histograms are very similar with respect to certain textons, however we also see that the primary fundamental textons that describe the terrain remain the same while the occurrence of these textons is less. We examine the likeness between the wet and dry terrains in Table 6.3 by exploring the histogram intersection between the two. The table show that while the terrains



Figure 6.7: Incorrect classification of the wetdirt3 sample. Here we see incorrect labeling to wet dirt, wet and dry pavement, and pine needles.

are mostly similar, there is at least a 10% difference between wet and dry samples that provide just enough uniqueness for the system to be able to accurately label the terrains separately.

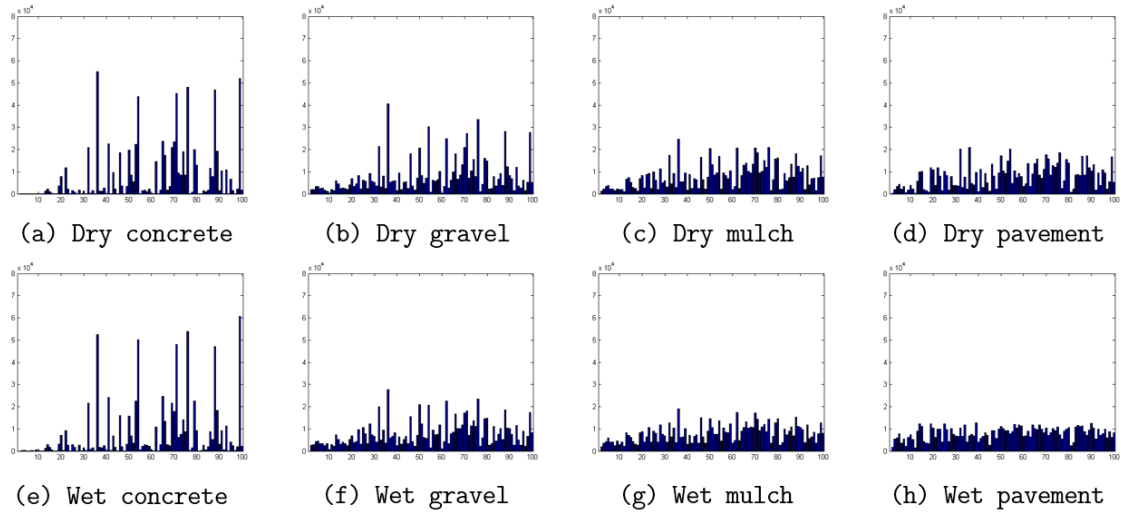


Figure 6.8: Comparison of dry and wet terrain histograms

Table 6.3: Histogram intersection for wet and dry terrains

Concrete	Gravel	Mulch	Pavement
80.03%	88.84%	89.75%	80.03%

Chapter 7

Conclusion and Future Work

In this thesis, we have proposed a new method for classifying terrains through multiview images. We designed and integrated a partially-autonomous mobile robot platform for acquiring ground terrains in an outdoor setting. We discussed novel ways for constructing texton dictionaries for clustering terrain information. Additionally, we have taken these texton dictionaries to build angular gradient differential histograms to classify a multitude of outdoor terrains with high accuracy. Finally, we have also found that the addition of a small angular shift in viewing direction to an angular gradient histogram provides a more robust means to classify terrains due to the effects of natural illumination.

There are many potential opportunities to continue this work in the future including an examination between different types of snow, ice, and snow ice mixtures. For example in 7.1 you can see the vast differences between specularities across each sample. It would be of great interest to fine tune autonomous robotic systems to account for the physical differences of these materials.

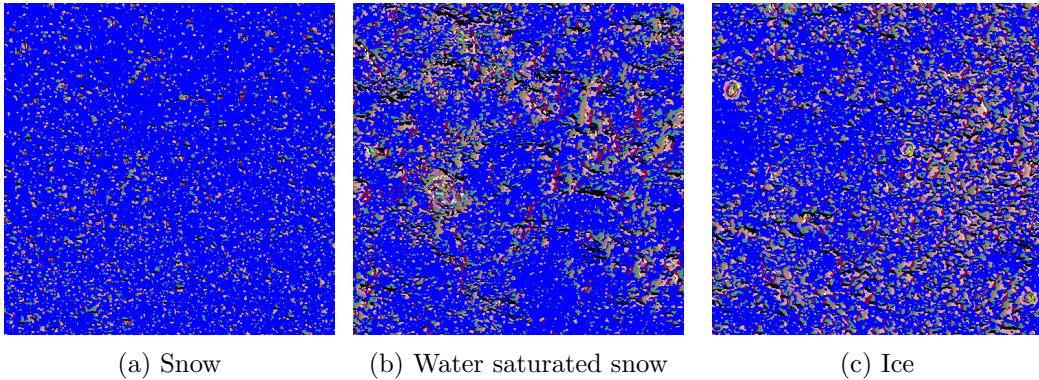


Figure 7.1: Comparison of cold weather related texton maps

Another extended area of work is to further examine the differences between terrains

that are both wet and dry. For example, the texton maps shown in Figure 7.2 are two samples from the same patch of concrete. However, Figure 7.2a is dry and Figure 7.2b has been saturated with water. Although they are both the same material and have the same composition, the effects of local environmental conditions dramatically bring out additional features for examination.

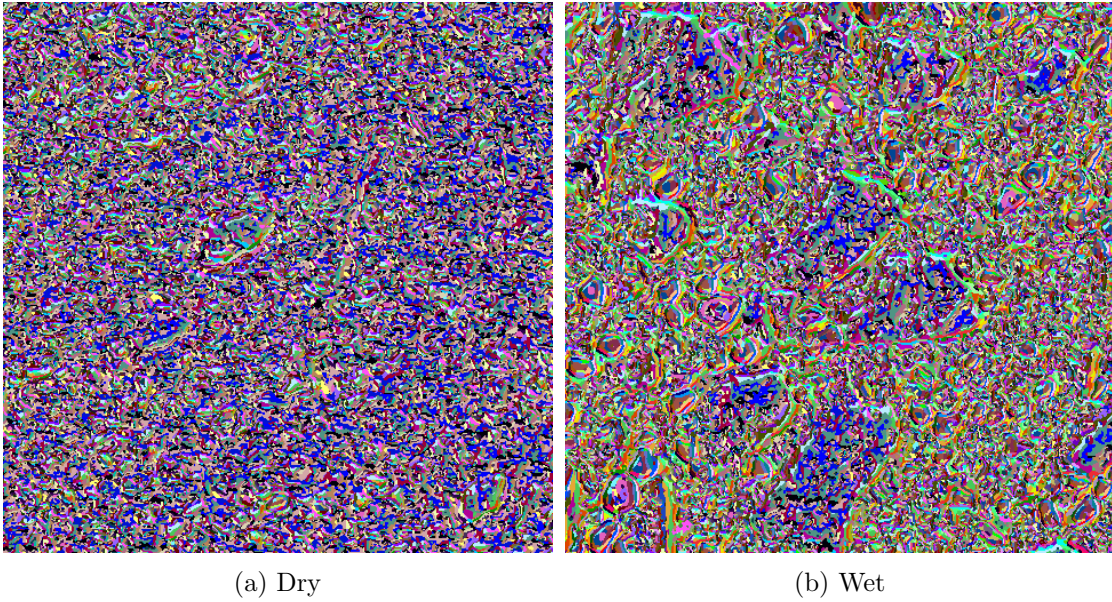


Figure 7.2: Comparison of texton maps for wet and dry samples of concrete

Appendix A

Table of Imaging Poses

Position Name	$Roll_a$	$Pitch_a$	Yaw_a	$Roll_b$	$Pitch_b$	Yaw_b
point1_1	0	45	90	0	5	180
point1_2	0	50	90	0	0	-180
point1_3	5	45	90	0	0	180
point2_1	0	60	90	0	0	180
point2_2	0	65	90	0	0	180
point2_3	5	60	90	0	0	180
point3_1	0	75	90	0	0	-180
point3_2	0	80	90	0	0	180
point3_3	5	75	90	0	0	180
point4_1	0	90	90	0	5	180
point4_2	0	95	90	0	45	180
point4_3	5	90	90	0	0	180
point5_1	-180	75	270	180	0	0
point5_2	-180	70	270	180	0	0
point5_3	-175	75	270	180	0	0
point6_1	-180	60	270	180	0	0
point6_2	-180	55	270	180	0	0
point6_3	-175	60	270	180	0	0
point7_1	-180	45	270	180	0	0
point7_2	-180	50	270	180	0	0
point7_3	-175	45	270	180	0	0

point8_1	-160	60	270	180	-5	0
point8_2	-160	65	270	-180	-45	0
point8_3	-165	60	270	180	0	0
point9_1	160	60	-90	-180	45	-360
point9_2	160	65	-90	-180	45	-360
point9_3	165	60	-90	-180	45	-360
point10_1	-20	60	90	0	45	-180
point10_2	-20	65	90	0	45	-180
point10_3	-15	60	90	0	45	-180
point11_1	20	60	90	0	-45	-180
point11_2	20	65	90	0	-45	-180
point11_3	15	60	90	0	-15	-180

Table A.1: Position and poses for camera placement. Arm positions with respect to the arm base are postfixed with a . Gripper positions with respect to the surface coordinate frame are postfixed with b .

Appendix B

Code for Automated Camera Control and Arm Movement

```

1  void CameraWidget::onRunProgramCode
2      (
3      )
4  {
5      // Automatically call PylonInitialize and
6      PylonTerminate to ensure the pylon runtime system
7      // is initialized during the lifetime of this object.
8      Pylon::PylonAutoInitTerm autoInitTerm;
9
10     try
11     {
12         // Only look for cameras supported by Camera_t
13         .
14
15         CDeviceInfo info;
16         info.SetDeviceClass(Camera_t::DeviceClass());
17
18         // Create an instant camera object with the
19         first found camera device that matches the
20         specified device class.
21
22         Camera_t camera(CTIFactory::GetInstance().
23             CreateFirstDevice(info));
24
25         // Open the camera to allow parameter changes

```

```

19         camera.Open();
20
21         // Turn test image off.
22         camera.TestImageSelector =
23             TestImageSelector_Off;
24
25         // Carry out luminance control by using the "
26             once" gain auto function.
27         AutoGainOnce(camera);
28
29         // Carry out white balance using the balance
30             white auto function.
31         AutoWhiteBalance(camera);
32
33         // Carry out luminance control by using the "
34             once" exposure auto function.
35         AutoExposureOnce(camera);
36
37         // Get the camera's nodemap to adjust
38             parameters via string
39         GenApi::INodeMap& nodemap = camera.GetNodeMap
40             ();
41
42         // Get any necessary camera nodes
43         GenApi::CEnumerationPtr triggerSelector(
44             nodemap.GetNode("TriggerSelector"));
45
46         GenApi::CEnumerationPtr triggerSource(nodemap.
47             GetNode("TriggerSource"));
48
49         GenApi::CEnumerationPtr triggerMode(nodemap.
50             GetNode("TriggerMode"));

```

```

40
41      // Set the camera up for software triggering
42      triggerSelector->FromString("FrameStart");
43      triggerMode->FromString("On");
44      triggerSource->FromString("Software");
45
46
47      // This smart pointer will receive the grab
         result data.
48      CGrabResultPtr ptrGrabResult;
49      String_t filename;
50
51      const QStringList manipActionList = m_pPlugin
         ->getManipActionList();
52      int numActions = (int) manipActionList.size();
53
54      for (int ii = 0; ii < numActions; ++ii)
55      {
56
57          m_pPlugin->runManipulationAction(
              manipActionList[ii].toString());
58
59          // Set up the stream grabber to start
              acquisition
60          camera.StartGrabbing();
61
62          // Execute a trigger so the camera can
              acquire
63          camera.ExecuteSoftwareTrigger();
64

```

```

65      // Wait for an image and then retrieve
        it. A timeout of 5000ms is used
66      camera.RetrieveResult(15000,
        ptrGrabResult,
        TimeoutHandling_ThrowException);
67
68      // Image grabbed successfully?
69      if (ptrGrabResult->GrabSucceeded())
70      {
71
72          /* The pylon grab result smart
            pointer classes provide a
            cast operator to the IImage
73          // interface. This makes it
            possible to pass a grab
            result directly to the
74          // function that saves an
            image to disk*/
75
76          }
77
78          //Save image
79          CImagePersistence::Save(
            ImageFileFormat_Bmp,
            filename, ptrGrabResult);
80
81      }
82
83

```

```
84         // end the command for the stream  
85         grabber to stop acquisition  
86         camera.StopGrabbing();  
87     }  
88  
89     camera.Close();  
90  
91     //Sample Completed  
92  
93     return 0;  
94 }
```

Appendix C

Imaging Procedure


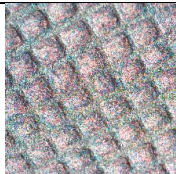
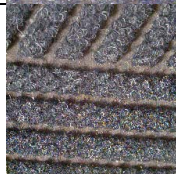





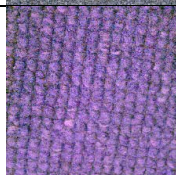
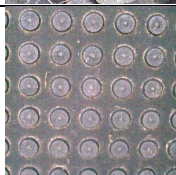
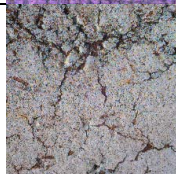

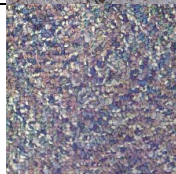

The procedure for collecting a single sample is as follows:

















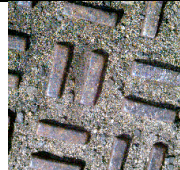



1. Turn on wireless network hotspot
2. Turn on robot (robot automatically connects to hotspot on initialization)
3. Connect laptop to mobile network
4. Run software for robot teleoperation
5. navigate robot to sample area of interest
6. Run software for arm manipulation
7. Load positions
8. Load poses
9. Initialize manipulation action sequence
10. Acquire 33 images as arm follows action sequence
11. Sample collection completes, navigate to next sample repeat from step 9

Overall, this process takes 15 minutes to complete per sample.

Appendix D

List of Database Samples

Sample Label	Image	Sample Label	Image
1.blackcarpet		26. redcarpet	
2. blackcarpet2		27. redtile	
3. blacktile		28. rocks	
4. blacktile2		29. rocks2	
5. bluecarpet		30. rubbermat	
6. streetpaint		31. sand	
7. browncarpet		32. slush1	

8. mixedcarpet		33. slush2	
9. drybrick		34. snow1	
10. dryconcrete		35. snow2	
11.dryconcrete2		36.wetconcrete	
12.dryconcrete3		37.wetconcrete2	
13. drygravel		38. wetdirt	
14. dryground		39. wetdirt2	
15. dryleaves		40. wetdirt3	
16. drymetal		41. wetdirt4	
17. drymulch		42.wetdirtwrocks	




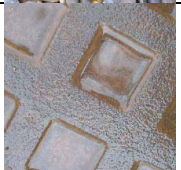








18. drypavement		43. wetgravel	
19. drypavement2		44. wetmetal	
20. grass		45. wetmulch	
21. grass2		46. wetpavement	
22. grass3		47. wetpavement2	
23. graycarpet		48. whitetile	
24. ice		49. whitetile2	
25. pineneedles		50. wood	

Table D.1: List of all collected samples and labels

References

- [1] E. H. Adelson. Image statistics and surface perception. In *Human Vision and Electronic Imaging XIII, Proceedings of the SPIE*, volume 6806, pages 680601–680609, 2008.
- [2] Adept. Pioneer 3-AT research robot platform. Pleasanton, CA.
- [3] Z. Akata, F. Perronnin, Z. Harchaoui, C. Schmid, Z. Akata, F. Perronnin, and Z. Harchaoui. Label-Embedding for Attribute-Based Classification. *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.
- [4] E. Angelopoulou, S. Lee, and R. Bajcsy. Spectral gradient: a material descriptor invariant to geometry and incident illumination. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, (January), 1999.
- [5] A. Angelova, L. Matthias, D. Helmick, G. Sibley, and P. Perona. Learning to predict slip for ground robots. *Proceedings - IEEE International Conference on Robotics and Automation*, 2006:3324–3331, 2006.
- [6] A. Angelova, L. Matthies, D. Helmick, and P. Perona. Slip Prediction Using Visual Information. *Proceedings of Robotics Science and Systems Conference*, 2006.
- [7] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(c):1–19, 2013.
- [8] Basler. Basler Industrial Cameras - ace Series - acA2040-90uc. Ahrensburg, Germany, 2014.
- [9] S. Bell, P. Upchurch, N. Snavely, and K. Bala. OPEN SURFACES : A Richly Annotated Catalog of Surface Appearance. *ACM Transactions on Graphics*, page 111, 2013.
- [10] M. Ben-Ezra, J. Wang, B. Wilburn, X. Li, and L. Ma. An LED-only BRDF measurement device. *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, (i), 2008.
- [11] M. Bibuli, M. Caccia, and L. Lapierre. Path-following algorithms and experiments for an autonomous surface vehicle. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 7(5):81–86, 2007.
- [12] S. Boughorbel, J. P. Tarel, and N. Boujemaa. Generalized histogram intersection kernel for image recognition. *Proceedings - International Conference on Image Processing, ICIP*, 3:161–164, 2005.

- [13] R. E. Broadhurst. Statistical estimation of histogram variation for texture classification. *Texture 2005 : Proceedings of the 4th International Workshop on Texture Analysis and Synthesis*,, pages 25–30, 2005.
- [14] G. Cai, B. M. Chen, and T. H. Lee. Unmanned Rotorcraft Systems. In *Coordinate Systems and Transformations*, Advances in Industrial Control, pages 23–35. Springer London, 2011.
- [15] F. S. Choong. A Gonioreflectometer for Measuring the Bidirectional Reflectance of Material for Use in Illumination Computation. *Thesis*, (August):145, 1997.
- [16] M. Cimpoi, S. Maji, and I. Kokkinos. Describing Textures in the Wild. *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.
- [17] E. Coyle, E. G. Collins, and L. Lu. Updating control modes based on terrain classification. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4417–4423, 2010.
- [18] O. G. Cula and K. J. Dana. Compact representation of bidirectional texture functions. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 1, 2001.
- [19] O. G. Cula and K. J. Dana. 3D texture recognition using bidirectional feature histograms. *International Journal of Computer Vision*, 59(1):33–60, 2004.
- [20] O. G. Cula and K. J. Dana. Texture for Appearance Models in Computer Vision and Graphics. In *Handbook of Texture Analysis*, number December, pages 1–23. 2008.
- [21] O. G. Cula, K. J. Dana, D. K. Pai, and D. Wang. Polarization multiplexing for bidirectional imaging. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:1116–1123, 2005.
- [22] O. G. Cula, K. J. Dana, D. K. Pai, and D. Wang. Polarization Multiplexing and Demultiplexing for Appearance-based Modeling. *PAMI*, 29:362–367, 2007.
- [23] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 886–893, 2005.
- [24] K. Dana, S. Nayar, B. V. Ginneken, and J. Koenderink. Reflectance and texture of real-world surfaces. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 18(1):1–34, 1997.
- [25] K. J. Dana and J. Wang. Device for convenient measurement of spatially varying bidirectional reflectance. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 21(1):1–12, 2004.
- [26] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 145–156, 2000.

- [27] J. Degraeve, R. V. Cauwenbergh, F. Wyffels, T. Waegeman, and B. Schrauwen. Terrain Classification for a Quadruped Robot. *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, (2):185–190, 2013.
- [28] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes Paris look like Paris? *ACM Transactions on Graphics*, 31(4):1–9, 2012.
- [29] R. O. Dror. Surface Reflectance Recognition and Real-World Illumination Statistics. *Dept. of Electrical Engineering and Computer Science*, page 150, 2002.
- [30] E. M. DuPont, C. a. Moore, E. G. Collins, and E. Coyle. Frequency response method for terrain classification in autonomous ground vehicles. *Autonomous Robots*, 24:337–347, 2008.
- [31] A. N. Erkan, R. Hadsell, P. Sermanet, J. Ben, U. Muller, and Y. LeCun. Adaptive long range vision in unstructured terrain. *IEEE International Conference on Intelligent Robots and Systems*, 1(1):2421–2426, 2007.
- [32] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object Detection with Discriminative Trained Part Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.
- [33] F. L. Garcia Bermudez, R. C. Julian, D. W. Haldane, P. Abbeel, and R. S. Fearing. Performance analysis and terrain classification for a legged robot over rough terrain. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 513–519, 2012.
- [34] P. Giguere. *Unsupervised Learning for Mobile Robot Terrain Classification*. PhD thesis, McGill University, 2010.
- [35] P. Giguere and G. Dudek. Clustering sensor data for autonomous terrain identification using time-dependency. *MIT Press*, pages 25–32, 2009.
- [36] R. Girshick, J. Donahue, T. Darrell, and J. Malik. object detection and semantic segmentation Detection & Segmentation. *ILSVRC 2013 Workshop*, 2013.
- [37] E. Guerra, J. de Lara, A. Malizia, and P. Díaz. Supporting user-oriented analysis for multi-view domain-specific visual languages. *Information and Software Technology*, 51(4):769–784, 2009.
- [38] E. Guizzo. How Google’s Self-Driving Car Works. *IEEE Spectrum*, Oct. 2011.
- [39] J. Y. Han and K. Perlin. Measuring bidirectional texture reflectance with a kaleidoscope. *ACM Transactions on Graphics*, 22(3):741, 2003.
- [40] K. Hara, K. Nishino, and K. Ikeuchi. Multiple light sources and reflectance property estimation based on a mixture of spherical distributions. *Proceedings of the IEEE International Conference on Computer Vision*, II:1627–1634, 2005.
- [41] K. Hara, K. Nishino, and K. Ikeuchi. Mixture of spherical distributions for single-view relighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1):25–35, 2008.

- [42] D. Hu and L. Bo. Toward Robust Material Recognition for Everyday Objects. *Proceedings of the British Machine Vision Conference*, pages 1–11, 2011.
- [43] B. Jeon, J. Kim, and J. Lee. Real-Time Recognition of the Terrain Configuration to Improve Driving Stability for Unmanned Robots. *World Academy of Science, Engineering and Technology International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering*, 8(2):269–274, 2014.
- [44] M. K. Johnson and E. H. Adelson. Retrographic sensing for the measurement of surface texture and shape. *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, pages 1070–1077, 2009.
- [45] R. E. Karlsen and G. Witus. Adaptive learning applied to terrain recognition. *Proceedings of SPIE*, 6962:69620H–69620H–10, 2008.
- [46] J. Kim, D. Kim, J. Lee, J. Lee, H. Joo, and I.-s. Kweon. Non-contact Terrain Classification for Autonomous Mobile Robot. *IEEE Int. Conf. on Robotics and Biomimetics*, pages 824–829, 2009.
- [47] G. J. Klinker, S. a. Shafer, and T. Kanade. The measurement of highlights in color images. *International Journal of Computer Vision*, 2(1):7–32, 1988.
- [48] J. J. Koenderink and A. J. van Doorn. Phenomenological description of bidirectional surface reflection. *Journal of the Optical Society of America A*, 15(11):2903, 1998.
- [49] C. Kyrkou and T. Theodoridis. A parallel hardware architecture for real-time object detection with support vector machines. *Computers, IEEE Transactions on*, 61(6):831–842, 2012.
- [50] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, pages 951–958, 2009.
- [51] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.
- [52] D. Levin. The cold, hard truth about autonomous vehicles and weather. *Fortune*, Feb. 2015.
- [53] W. Li, Q. Yu, H. Sawhney, and N. Vasconcelos. Recognizing Activities via Bag of Words for Attribute Dynamics. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2587–2594, 2013.
- [54] T. Litman. Autonomous Vehicle Implementation Predictions: Implications for Transport Planning. *Transportation Research Board Annual Meeting*, pages 36–42, 2014.

- [55] C. Liu and J. Gu. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral BRDF. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):86–98, 2014.
- [56] C. Liu, L. Sharan, E. H. Adelson, and R. Rosenholtz. Exploring features in a Bayesian framework for material recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 239–246, 2010.
- [57] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. SIFT flow: dense correspondence across difference scenes. In *Computer Vision ECCV 2008*, Lecture Notes in Computer Science, pages 28–42. Springer Berlin Heidelberg, 2008.
- [58] S. Lombardi and K. Nishino. Single Image Multimaterial Estimation. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 238–245, 2012.
- [59] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, pages 1–28, 2004.
- [60] J. Masci, U. Meier, D. Cirean, and J. Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6791 LNCS(PART 1):52–59, 2011.
- [61] S. Mattoccia, I. Marchio, and M. Casadio. A Compact 3D Camera Suited for Mobile and Embedded Vision Applications. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3–4, 2014.
- [62] K. Mizuno, Y. Terachi, K. Takagi, S. Izumi, H. Kawaguchi, and M. Yoshimoto. Architectural study of HOG feature extraction processor for real-time object detection. *IEEE Workshop on Signal Processing Systems, SiPS: Design and Implementation*, pages 197–202, 2012.
- [63] S. Nayar, K. Ikeuchi, and T. Kanade. Shape from interreflections. *[1990] Proceedings Third International Conference on Computer Vision*, pages 2–11, 1990.
- [64] S. K. Nayar, K. Ikeuchi, and T. Kanade. Surface reflection: Physical and geometrical perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):611–634, 1991.
- [65] F. Nicodemus, J. Richmond, and J. Hsia. Geometrical considerations and nomenclature for reflectance. *Science And Technology*, 60(October):1–52, 1977.
- [66] K. Nishino. Directional statistics BRDF model. *Proceedings of the IEEE International Conference on Computer Vision*, pages 476–483, 2009.
- [67] K. Nishino and K. Ikeuchi. Robust simultaneous registration of multiple range images. *Digitally Archiving Cultural Objects*, (January):71–88, 2008.
- [68] K. Nishino and S. Lombardi. Directional statistics-based reflectance model for isotropic bidirectional reflectance distribution functions. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 28(1):8–18, 2011.

- [69] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.
- [70] G. Oxholm, P. Bariya, and K. Nishino. The scale of geometric texture. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7572 LNCS(PART 1):58–71, 2012.
- [71] G. Oxholm and K. Nishino. Shape and Reflectance from Natural Illumination. *Computer Vision – ECCV 2012*, 7572:528–541, 2012.
- [72] P. Pachowicz. Learning-based architecture for robust recognition of variable texture to navigate in natural terrain. *EEE International Workshop on Intelligent Robots and Systems, Towards a New Frontier of Applications*, pages 135–142, 1990.
- [73] G. Patterson and J. Hays. SUN attribute database: Discovering, annotating, and recognizing scene attributes. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2751–2758, 2012.
- [74] O. Pele and M. Werman. The quadratic-chi histogram distance family. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6312 LNCS(PART 2):749–762, 2010.
- [75] M. Ranzato, Fu Jie Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition. *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07.*, pages 1–8, 2007.
- [76] M. Reinstein, V. Kubelka, and K. Zimmermann. Terrain adaptive odometry for mobile skid-steer robots. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4706–4711, 2013.
- [77] Robai. Cyton Gamma 300 Arm Specifications. Cambridge, MA, 2014.
- [78] S. M. Rusinkiewicz. A New Change of Variables for Efficient BRDF Representation. *Rendering Techniques, Proceedings of Eurograph - ics Rendering Workshop*, pages 11–22, 1998.
- [79] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008.
- [80] D. Sadhukhan. *Autonomous Ground Vehicle Terrain Classification Using Internal Sensors*. PhD thesis, The Florida State University DigiNole Commons, 2004.
- [81] R. Sagawa, K. Nishino, and K. Ikeuchi. Adaptively merging large-scale range data with reflectance properties. *Digitally Archiving Cultural Objects*, 27(3):161–191, 2008.

- [82] Y. Sato, M. D. Wheeler, and K. Ikeuchi. Object shape and reflectance modeling from observation. *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 379–388, 1997.
- [83] A. Saudabayev, F. Kungozhin, D. Nurseitov, and H. A. Varol. Depth Image based Terrain Recognition for Supervisory Control of a Hybrid Quadraped. In *IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pages 1532–1537, 2014.
- [84] L. Sharan, C. Liu, R. Rosenholtz, and E. H. Adelson. Recognizing materials using perceptually inspired features. *International Journal of Computer Vision*, 103(3):348–371, 2013.
- [85] J. Shotton, J. Winn, C. Rother, and A. Criminisi. TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. In *Computer Vision ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 1–15. 2006.
- [86] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor Segmentation and Support Inference from RGBD Images. *Computer Vision–ECCV*, pages 746–760, 2012.
- [87] B. Sofman, J. A. D. Bagnell, A. T. Stentz, and N. Vandapel. Terrain Classification from Aerial Data to Support Ground Vehicle Navigation. Technical report, 2006.
- [88] J. Sun, J. M. Rehg, and A. F. Bobick. Learning for Ground Robot Navigation with Autonomous Data Collection. *Learning*, 2005.
- [89] R. T. Tan, K. Nishino, and K. Ikeuchi. Separating reflection components based on chromaticity and noise analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1373–1379, 2004.
- [90] S. Tominaga and B. a. Wandell. Standard surface-reflectance model and illuminant estimation. 6(4):576–584, 1989.
- [91] a. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2, 2004.
- [92] M. Varma and A. Zisserman. Classifying images of materials: Achieving view-point and illumination independence. *Computer Vision ECCV 2002*, pages 255–271, 2002.
- [93] M. Varma and A. Zisserman. Texture Classification : Are Filter Banks Necessary? *Computer Vision and Pattern Recognition 2003 Proceedings 2003 IEEE Computer Society Conference on*, 2003.
- [94] M. Varma and A. Zisserman. A Statistical Approach to Texture Classification from Single Images. *International Journal of Computer Vision*, 62:61–81, 2005.
- [95] B. Vijayalakshmi and V. S. Bharathi. A Novel Approach to Texture Classification using Statistical Feature. *Journal of Information Technology*, 1(5):31–39, 2011.

- [96] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. Inverting and visualizing features for object detection. Technical report, 2012.
- [97] K. Walas. Terrain Classification and Negotiation with a Walking Robot. *Journal of Intelligent & Robotic Systems*, (May), 2014.
- [98] K. Walas, A. Schmidt, M. Kraft, and M. Fularz. Hardware implementation of ground classification for a walking robot. *9th International Workshop on Robot Motion and Control, RoMoCo 2013 - Workshop Proceedings*, pages 110–115, 2013.
- [99] J. Wang and K. J. Dana. Relief texture from specularities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):446–457, 2006.
- [100] J. W. J. Wang and K. Dana. Hybrid textons: modeling surfaces with reflectance and geometry. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 1, 2004.
- [101] G. J. Ward. Measuring and modeling anisotropic reflection. *ACM SIGGRAPH Computer Graphics*, 26(2):265–272, 1992.
- [102] J. S. Weszka, C. R. Dyer, and A. Rosenfeld. Comparative Study of Texture Measures for Terrain Classification. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-6(4):269–285, 1976.
- [103] T. Weyrich, W. Matusik, H. Pfister, B. Bickel, C. Donner, C. Tu, J. McAndless, J. Lee, a. Ngan, H. Jensen, and M. Gross. Analysis of Human Faces using a Measurement-Based Skin Reflectance Model. *ACM Transactions on Graphics*, 25(3):1013, 2006.
- [104] J. Winn, a. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. *Proceedings of the IEEE International Conference on Computer Vision*, II:1800–1807, 2005.
- [105] D. F. Wolf, G. S. Sukhatme, D. Fox, and W. Burgard. Autonomous Terrain Mapping and Classification Using Hidden Markov Models. *International Conference on Robotics and Automation*, (April):2026–2031, 2005.
- [106] J. Xiao, K. a. Ehinger, A. Oliva, and A. Torralba. Recognizing scene viewpoint using panoramic place representation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2695–2702, 2012.
- [107] J. Xiao, A. Owens, and A. Torralba. SUN3D: A database of big spaces reconstructed using SfM and object labels. *Proceedings of the IEEE International Conference on Computer Vision*, pages 1625–1632, 2013.
- [108] F. Zhang, Z. Fang, M. Liu, and H. Huang. Preliminary design of a terrain recognition system. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pages 5452–5455, 2011.
- [109] H. Zhang, K. Dana, and K. Nishino. Reflectance Hashing for Material Recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

- [110] H. Z. H. Zhang, D. O. D. Ouyang, H. L. H. Lin, and W. G. W. Guan. Texture Synthesis Based on Terrain Feature Recognition. *2008 International Conference on Computer Science and Software Engineering*, 2:1158–1161, 2008.
- [111] J. Zhu, Y. Wang, H. Yu, H. Xu, and Y. Shi. Obstacle detection and recognition in natural terrain for field mobile robot navigation. *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, pages 6567–6572, 2010.