

University of Kentucky UKnowledge

University of Kentucky Master's Theses

Graduate School

2005

DESIGN OF A GAIT ACQUISITION AND ANALYSIS SYSTEM FOR ASSESSING THE RECOVERY OF MICE POST-SPINAL CORD INJURY

Casey Harr University of Kentucky, cpharr0@uky.edu

Right click to open a feedback form in a new tab to let us know how this document benefits you.

Recommended Citation

Harr, Casey, "DESIGN OF A GAIT ACQUISITION AND ANALYSIS SYSTEM FOR ASSESSING THE RECOVERY OF MICE POST-SPINAL CORD INJURY" (2005). *University of Kentucky Master's Theses*. 262. https://uknowledge.uky.edu/gradschool_theses/262

This Thesis is brought to you for free and open access by the Graduate School at UKnowledge. It has been accepted for inclusion in University of Kentucky Master's Theses by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.

ABSTRACT OF THESIS

DESIGN OF A GAIT ACQUISITION AND ANALYSIS SYSTEM FOR ASSESSING THE RECOVERY OF MICE POST-SPINAL CORD INJURY

Current methods of determining spinal cord recovery in mice, post-directed injury, are qualitative measures. This is due to the small size and quickness of mice. This thesis presents a design for a gait acquisition and analysis system able to capture the footfalls of a mouse, extract position and timing data, and report quantitative gait metrics to the operator. These metrics can then be used to evaluate the recovery of the mouse. This work presents the design evolution of the system, from initial sensor design concepts through prototyping and testing to the final implementation. The system utilizes a machine vision camera, a well-designed walkway enclosure, and image processing techniques to capture and analyze paw strikes. Quantitative results gained from live animal experiments are presented, and it is shown how the measurements can be used to determine healthy, injured, and recovered gait.

KEYWORDS: Mouse Gait, Machine Vision Camera, Hough Transform, BBB Score, Spinal Cord Recovery

> Casey Harr December 15, 2005

DESIGN OF A GAIT ACQUISITION AND ANALYSIS SYSTEM FOR ASSESSING THE RECOVERY OF MICE POST-SPINAL CORD INJURY

By

Casey Patrick Harr

James E. Lumpp Jr. Director of Thesis Yu-Ming Zhang Director of Graduate Studies

December 15, 2005

RULES FOR THE USE OF THESES

Unpublished theses submitted for the Master's degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgments.

Extensive copying or publication of the thesis in whole or in part also requires the consent of the Dean of the Graduate School of the University of Kentucky.

A library that borrows this thesis for use by its patrons is expected to secure the signature of each user.

THESIS

Casey Patrick Harr

The Graduate School University of Kentucky 2005

DESIGN OF A GAIT ACQUISITION AND ANALYSIS SYSTEM FOR ASSESSING THE RECOVERY OF MICE POST-SPINAL CORD INJURY

THESIS

A thesis submitted in partial fulfillment of the requirements of the degree of Masters of Science in the College of Engineering at the University of Kentucky

By

Casey Patrick Harr

Lexington, KY

Chair: Dr. James E. Lumpp Jr.,

Associate Professor of Electrical and Computer Engineering

Lexington, KY

2005

Copyright © Casey Patrick Harr 2005.

To Grandma and Grandpa Roderick for always listening to my crazy ideas.

ACKNOWLEDGMENTS

Although this project was an individual endeavor, there are those from whom I gained insight and direction, and therefore must be properly acknowledged. First and foremost, I thank my Thesis Chair, Dr. James E. Lumpp, for allowing me the opportunity to conduct this research and to pursue my Masters of Science degree. He has provided me with scholarship and encouragement through all the years we've been associated. I also wish to thank my committee members, Dr. Steven Scheff, and Dr. Ingrid St. Omer. Dr. Scheff's instructive feedback was the driving force behind this project's development and Dr. St. Omer's flexibility during the defense process allowed for a higher quality of the final Thesis. She even gave up her lunch to sit for the defense.

To collect the live animal experimental data presented in this Thesis, I relied on the assistance of Kelly Miller and Kelly Roberts. Having to use the system, these two girls provided the most intuitive design comments. They also demonstrated a great amount of patience as the system often did not perform as it should have. Their sense of humor was appreciated as well.

In addition to those mentioned above, I'd also like to thank everyone who worked with me in the lab. This includes Osamah Rawashdeh, David Jackson, Adam Groves, Darren Brown, Tim Arrowsmith, and Garrett Chandler. They helped by letting me to bounce ideas off them, giving a outside look at my designs, and listening to me complain. I might have gotten work done faster without them around, but it wouldn't have been as enjoyable.

Lastly, I'd like to acknowledge my parents, Ken and Jill Harr. It is their constant support that allows me to achieve whatever goal I desire. They done raised me right.

Also, Crackers is the best mouse of them all.

TABLE OF CONTENTS

ACKNOV	WLEDGMENTS iii	
LIST OF FIGURES vii		
LIST OF	TABLES ix	
CHAPTE	R 1: INTRODUCTION1	
1.1	Background1	
1.2	Objective	
1.3	Thesis Overview	
CHAPTE	R 2: SURVEY OF TECHNOLOGIES	
2.1	Introduction	
2.2	Resistive Touch Sensors	
2.3	Capacitive Touch Sensors	
2.4	Acoustic Touch Sensors	
2.5	Fiber Optic Touch Sensors	
2.6	Piezoelectric Polymer Film	
2.7	Machine Vision Cameras	
2.8	Kistler Force Plate	
2.9	Fingerworks Touchpad	
2.10	Pressure Profile Systems TactArray	
2.11	Tekscan Animal Gait Measurement System	
2.12	Mouse Specifics DigiGait Imaging System	
2.13	Discussion	
СНАРТЕ	R 3: TECHNOLOGY FEASIBILITY EXPERIMENTS	
3.1	Introduction	
3.2	Addressing Schemes	
3.3	Dual Laver Piezoelectric Film Array	
3.4	Capacitive Sensor Array	
3.5	Camera/Plate System	
3.6	Discussion	
CHAPTE	R 4: GAIT ACOUISITION AND ANALYSIS DEVICE SPECIFICATION . 34	
4.1	Introduction	
4.2	System Requirements	
4.3	Design Overview	
CHAPTE	R 5: MACHINE VISION CAMERA	
5.1	Introduction	
5.2	Capabilities	

5.3 \$	Software Development Kit	39
5.4 I	_imitations	40
5.5 I	Lenses	41
CHAPTER	6: WALKWAY STAND AND DEVICE HARDWARE	43
6.1 I	ntroduction	43
6.2 0	Components	43
6.2.1	Walkway Plates	44
6.2.2	2 Track Enclosure	46
6.2.3	3 Lighting	47
6.2.4	External Hardware	49
6.2.5	6 PC Capture Software	
6.3 I	Prototypes	53
6.3.	"Pagoda" Proof-of-Concept Testbed	53
6.3.4	2 "Poplar" Device Parameters Testbed	
0.3.3	Fleet Foot Complete System Implementation	05
0.3.4	Fleet Foot Complete System Implementation - Mark II	14
0.4 I	Physical Dimensions	//
6.4.1	I Flysical Dimensions	70
	7. IMAGE DEOCESSING TECHNIQUES	70
	7. IMAGE PROCESSING TECHNIQUES	79
7.1 I 7.2 I	ocating Relevant Data	79
7.2 1	Separation by Threshold	79
7.2.3	P Edge Detection	/)
7.2.3	Bifference of Gaussian	
7.3	Categorizing Data	88
7.3.1	Shape Clustering	
7.3.2	2 Parametric Descriptor	90
7.3.3	Robust Key Points	92
7.4 (Dbject Detection and Identification	92
7.4.1	Sum of Absolute Differences	93
7.4.2	2 Cross-Correlation	93
7.4.3	3 Shape Number	93
7.4.4	Hough Transformation	95
CHAPTER	8: PAW IDENTIFICATION ALGORITHM	98
8.1 I	Detection Requirements	
8.2 I	mplementation	100
8.2.]	Descriptor Matching Based on Centroid of Difference of Gaussian	100
l o o o	Kegions	100
8.2.2	Locally Adaptive Generalized Hough Transform	102
8.3 I	Inal Implementation	109

CHAPTER 9: H	EXPERIMENTAL RESULTS	111
9.1 Intro	oduction	111
9.2 Live	e Animal Experiments	111
9.2.1 H	Healthy White Mice	112
9.2.2 I	Injured White Mice	114
9.2.3 H	Healthy Black Mice	115
9.2.4 I	Injured Black Mice	117
9.3 Hit/N	Miss/False Detection Ratio	118
9.4 Mice	e Training	120
CHAPTER 10:	CONCLUSION	121
10.1 Intro	oduction	121
10.2 Gait	t Data Reported	121
10.3 Anal	lysis of Gait Data	122
10.4 Trac	cking Recovery Using Gait Data	127
10.5 Com	nparison to BMS Score	129
10.6 Refi	inement of the System	133
10.6.1 H	Refinement of Camera/Plate Stand	133
10.6.2 H	Refinement of Detection Algorithm	134
10.6.3 A	Alternative Designs	135
10.7 Fina	al Design Review	136
Appendix I		137
References		138
Vita		141

LIST OF FIGURES

Figure 1. Live Animal Response of Fingerworks Touchpad	. 14
Figure 2. DigiGait Analysis Screen	. 16
Figure 3. Dual Layer Bus Addressing	. 19
Figure 4. Passive Element Matrix Addressing	. 21
Figure 5. Active Element Matrix Addressing	. 22
Figure 6. Piezoelectric Film Sensor	. 23
Figure 7. Dual Layer Piezoelectric Touch Sensor Test	. 23
Figure 8. Response of Dual Layer Piezoelectric Sensor to Touch	. 24
Figure 9. QProx Single and Dual Input QTouch Boards	. 26
Figure 10. QProx E6645 Evaluation Kit	. 27
Figure 11. QT60486 Sensor Chip Circuit Diagram	. 28
Figure 12. Interdigitated Capacitive Sensor Grid Layout	. 29
Figure 13. Logitech Quickcam PC Camera for Notebooks	. 31
Figure 14. PixeLINK PL-A741-R Machine Vision Camera	. 37
Figure 15. Screen Capture of PixeLINK Capture OEM	. 39
Figure 16. Comparison of Descending Imaging Resolutions	. 41
Figure 17. Lens Geometry	. 42
Figure 18. Clear plastic walkway and frosted glass walkway	. 44
Figure 19. Frame Color Filtering Tests	. 45
Figure 20. Original and Refined Mouse Houses	. 46
Figure 21. Rectangular and Triangular Canopies	. 47
Figure 22. Lighting Sources Tested	. 48
Figure 23. MOSFET Based Triggering Circuit	. 50
Figure 24. Revised Microcontroller Based Triggering Circuit	. 51
Figure 25. Pagoda Camera/Plate Test Stand	. 54
Figure 26. Successive Frames Captured at 15 fps	. 56
Figure 27. Poplar Gait Stand	. 57
Figure 28. Poplar Stand Lighting Mounts	. 58
Figure 29. Optical Blurring Effect Due to Frosted Glass	. 61
Figure 31. Fleet Foot Gait Acquisition System	. 66
Figure 32. Intensity "Fish-Eye" Effect	. 66
Figure 33. Light Mouse House with Translucent Floor	. 67
Figure 34. Walkway Insertion Channels	. 69
Figure 35. Modular Halves of Fleet Foot Stand	. 70
Figure 36. Paws Washed Out and in Dark Edge Region	. 71
Figure 37. Mouse with Inked Paws Before and After Injury	. 73
Figure 38. The Fleet Foot Gait Acquisition System - Mark II	. 74
Figure 39. Fleet Foot Mark II Enclosure Modifications	. 75
Figure 40. Dual Level Lighting	. 76
Figure 41. Dual Level and Single Level Lighting Comparison	. 77
Figure 42. Basic Thresholding at Incremental Levels	. 80

Figure 43. Foreground Extraction Using Background Normalization	81
Figure 44. Sobel and Prewitt Edge Detection	82
Figure 45. Canny and Zero-Cross Edge Detection Results at Varying Thresholds	85
Figure 46. Four Levels of Difference of Gaussian Filtered Image	88
Figure 47. Grouping Pixel Regions by Proximity	89
Figure 48. Dual Pass Grouping by Connectivity	90
Figure 49. Parametric Descriptors Based on a Central Reference Point and	
Interconnection	92
Figure 50. Computing an N4 Shape Number (Chain Code)	94
Figure 51. Generalized Hough Transform Descriptor Parameters	96
Figure 52. Multiple Paw Contact Configurations	98
Figure 53. Flaws in the Image Data	99
Figure 54. Pixel Clustering of DoG Data	. 101
Figure 55. Uniformity Mask, Raw Frame, and Background Normalized Frame	. 103
Figure 56. Left and Right Hind Paws Samples Used to Generate Hind Paw	
Descriptors	. 104
Figure 57. Master Target Descriptor Image for White Mouse Hind Right Paw	. 105
Figure 58. Descriptors Based on Variation in Paw Poses	. 105
Figure 59. Healthy White Mouse with Paw Identification Markers	. 112
Figure 60. Gait Trace of Healthy White Mouse	. 113
Figure 61. Injured White Mouse with Paw Identification Markers	. 114
Figure 62. Gait Trace of Injured White Mouse	. 115
Figure 63. Healthy Black Mouse with Paw Identification Markers	. 116
Figure 64. Gait Trace of Healthy Black Mouse	. 117
Figure 65. Injured Black Mouse with Paw Identification Markers	. 117
Figure 66. Gait Trace of Injured Black Mouse	. 118
Figure 67. Healthy Animal Gait Traces Showing Interlimb Coordination	. 125
Figure 68. Gait Traces Showing Recovery of Black Mouse Red-2	. 128
Figure 69. BM Red-2 After Partial Injury Recovery	. 129

LIST OF TABLES

Table 1. Machine Vison Camera Comparison	30
Table 2. Quickcam Working Distance and Resolution Comparison.	55
Table 3. Raw Gait Data Generated by the Identification Algorithm	121
Table 4. Computed Gait Metrics Based on Acquired Data	122

1 INTRODUCTION

1.1 Background

Current medical research is being conducted with the intent of understanding the intricacies of spinal cord injury. Though the ability to perform precision surgery on test animals is highly advanced [1], [2], [3], the methods used to ascertain recovery are still qualitative measures. Lab mice play a particularly integral role in spinal trauma research, but due to their small size and quickness, the results of the current means of measure, the BBB score, or the reduced BMS score, are highly subjective.

The BBB score was developed by Michele Basso, Michael Beattie, and Jacqueline Bresnahan at Ohio State University in 1995 [4]. The BBB score is a 21 point scale used to rate the recovery of rats post spinal trauma. The scale is applied by having a trained operator observe the injured animals and score them based on specific behaviors associated with trauma recovery. The BBB score is separated into three categories, with seven rankings per category, based on the key phases of recovery. The first phase, early recovery, scores the animal's hindlimb joint movement, ranging from no movement to extensive movement of all three joints on each side. The second phase, intermediate recovery, scores paw placement (plantar stepping), weight support, and interlimb coordination. The final phase, late recovery, scores paw rotation, tail support, toe dragging, and body stability. It is assumed that for each ranking, all the requirements of the previous ranks are met. A healthy or fully recovered animal will walk with full interlimb coordination, plantar steps, weight support, paws rotated parallel to the body, no toe dragging, tail up off the ground, and stable body posture.

The BBB score can be used to rate the recovery of mice, as well as rats, but due to the small size and increased quickness of mice, a reduced scale, the BMS score (Basso Mouse Scale), has been developed [5]. The BMS score is a 9 point scale were granularity is decreased by removing recovery criterion which are too difficult to ascertain for mice. Ratings based on

individual joint movement, toe clearance, or finer degrees of particular recovery criteria are removed. The scale is still separated into the three recovery phases: early, intermediate, and late. Early recovery rates basic ankle movement. Intermediate recovery rates occasional plantar stepping with limited weight support. Late recovery rates coordinated stepping, paw rotation, body stability, and tail support. Detailed criteria for each scale point is given in Appendix I.

The BBB and the BMS scores are able to provide researchers some level of agreement between measures of mouse recovery, but there remain several shortcomings. The scoring is inherently subjective [6], [7], being based on the adeptness of the observer. The observers must be trained to identify particular characteristics and uniformly score each gradual change in behavior. Even with adequate training, scores can vary slightly from one observer to the next. The results can also be easily misinterpreted. The difference between 3 and 4 on the BMS scale is not as significant as the difference between 4 and 5 which reflects a shift in recovery phase. Each phase represents an important biological change and this method of scoring does not highlight those transitions. These limitations show that a more reliable method of recovery tracking is needed; a method by which animal recovery can be determined by quantifiable measurement data.

The average healthy mouse has a mass of approximately 30g. It moves at an average velocity of 26 cm/s within a range of 14 cm/s to 43 cm/s. The average amount of time spent per complete stride is 260ms where the forelimbs spend an average of 160ms in contact with the ground and the hindlimbs an average of 185ms. The average length of a stride is 6.76cm and the average stance width is 2.7cm. The paws are approximately 1 cm^2 for the fore paws and $1 \text{ cm} \times 1.5 \text{ cm}$ for the hind paws. They typically contact the ground at a 5° rotation outward from the center line. For a normal gait, the hind paw will land just behind where the previous fore paw was. Mice move with coordinated stepping, meaning that for every forelimb step taken, a hindlimb step of the alternating side follows. Therefore, alternate-side, forelimb-hindlimb pairs move synchronously. [8], [9], [10].

1.2 Objective

The objective of this work is to develop a complete system which will capture the movement of a mouse and process it to derive quantifiable data describing the dynamics of the gait. These gait metrics can then be analyzed before and after a spinal injury has occurred to track the progress of recovery treatments. The device will also have archival and retrieval functionality to allow the review of experiments at each stage of the recovery cycle. The goal is to provide an analytical alternative to the methods presently being employed to rank animal recovery.

The functional requirements of the gait analysis device stem from three distinct groups; engineering, operability, and data reporting. The device first and foremost must have a high sensor resolution and sample rate. The minimum resolution is 10 sensor nodes per inch. This will show where a foot has made contact but will not provide any clarity. The ideal resolution is 100 sensor nodes per inch or greater. This will give enough data per foot to accurately display the separation of digits and the rotation. The minimum sample rate is 40 frames per second. This correlates to 10 frames of data per full stride. The ideal sample rate is 100 frames per second or greater. This provides 25 frames per full stride on average which is enough to get decent timing measurements.

The force applied at each sensor node must also be measured. The device has to sense a minimum of 1g per node to have enough resolution to detect if the animal is favoring one limb over another. The total sensing area of the device must be large enough to accumulate enough data to compute average measurements. The device must also sense multiple contact points as more than one paw may be in contact with the surface at a time.

The operability requirements pertain to the use of the device with live animals in a medical lab setting. It must be physically robust, tolerant to repeat usage, tolerant to animal excretions, and be quick and easy to clean. The system must be completely self contained, with the exception of a PC interface. Lastly, it is required that no preparation of the animal is necessary for the device to be used. Training of the animals in advance, though, is acceptable. Applying ink to the animals' feet is permissible, though undesirable.

The data reported to the user must identify as many of the following gait metrics as possible. The positional metrics include stride length, stance width, coordination, maximum contact area, and paw rotation. The temporal metrics include stride duration, stance duration, swing duration, and tail dragging duration. Additional metrics include inter-limb coordination, stride frequency, velocity, paw pressure, and ground reaction force.

1.3 Thesis Overview

This thesis will explore two methods of acquiring gait data: the development of a solid-state sensor array and the development of a camera-based solution. Each design method has its advantages and disadvantages which will be discussed. Ultimately, it will be shown that the camera-based design is the most practical to implement. The evolution of the image processing algorithm used to identify paw placement within a data frame will also be discussed.

The bulk of this thesis will discuss design challenges and the related solutions leading to the final system design. There were three major challenges encountered during the research process. The first being the ability to manufacture a prototype of the solid-state sensor array. To obtain even a minimal amount of resolution, a prohibitively large number of sensing elements had to be manufactured at a prohibitively small size. Research was conducted to identify a suitable addressing scheme to access the large number of elements easily, but, as will be shown, nothing was found to be sufficient. This directly led to the decision to switch the system to a camera-based model. The second challenge dealt with the complexity involved in lighting the gait capture area to create enough contrast for the camera to

distinguish between the mouse's paws the rest of the body and background. A variety of lighting schemes were implemented to attempt to force the intensity of the feet to be different enough for reliable detection by the image processing algorithm. The third challenge was developing an image matching algorithm which could parse through the large amounts of video data in a timely manner with a minimal number of false or missed matches.

The main focus of this thesis is the development of the acquisition and analysis system, including all aspects of hardware design, interfacing, and software development. Some work has been done to gauge the effectiveness of this system for replacing the current method of determining spinal trauma recovery, but only within the scope of improving the engineering design.

The first chapter of this thesis presents the current method used to assign value to the level of recovery of mice post spinal cord trauma. A brief introduction to the gait dynamics of an average healthy mouse is then provided. The objective of the thesis is then outlined with a synopsis of the basic functional requirements and the challenges the design presents. Lastly, an overview of the thesis chapters is given.

In the second chapter, a variety of technologies are surveyed which demonstrate the potential for use as the core sensor element of the device. These technologies include several solidstate sensor types, pre-existing touchscreen technologies, and digital imaging cameras. Additionally, there are several existing device which function similarly to the propose system or may be adapted for use in some way. The chapter ends with a discussion of the limitations of each technology and which deserve further consideration.

The third chapter presents the initial feasability experiments conducted with the three most promising sensor technologies. The chapter begins by briefly discussing addressing schemes which were considered for implementing sensor arrays. The three sensor test devices, the dual layer piezoelectric film array, the capacitive sensor array, and the camera/plate system are then reviewed. Lastly, the merits and limitations of the three systems are presented. The

conclusion to use the camera/plate system is made after detailing the design restrictions of the other two technologies. This also requires a refinement of the overall system requirement to be made.

The forth chapter refines the general problem statement to a concise design based upon the camera/plate sensing system. The chapter begins by detailing the progression of development of the camera/plate system. Next, the specific design criteria of the gait acquisition and analysis device are set. Lastly, a brief overview is presented of the individual components of the device which must be developed and what potential design challenges may arise.

In the fifth chapter, the machine vision camera to be used for the final device is chosen. The chapter begins by providing a brief description of the camera. Next, the major capabilities of the camera are described in detail. The software development kit used to access camera features and retrieve the image data is then discussed. The limitations of the camera are then documented and the challenges which will need to be solved during the development process are discussed. Lastly, a discussion on choosing an appropriate lens for the application is presented.

Chapter six, which documents the device design evolution, is one half of the bulk of this research. Focusing only on the physical stand design, this chapter documents the evolution of the hardware from a basic camera and plastic pane to a fully functional experimental device. It is important to see the maturation of the physical hardware to understand why each component included in the final design is critical to overall device functionality. This chapter is broken into too main parts. The first documents each component group (e.g. walkway plates, lighting, enclosure). The second presents the prototype design sequence and highlights the changes made to each successive stage. The chapter ends by giving a complete description of the final physical design.

Chapters seven and eight represent the other major half of the research, the image processing and paw identification algorithm. Chapter seven presents the image processing techniques explored. The general use of each of these methods are explained in detail and the specific use for each of them pertaining to paw identification is discussed.

Chapter eight presents the two paw detection algorithms which were developed. A full description of each is given, along with the details of each image processing technique used during each stage of the algorithm. The results and limitations of each algorithm are also presented, including why the first algorithm was abandoned in lieu of the second one.

Chapter nine presents the experimental results acquired from the final system design. The results are reported for both healthy and injured animals. These results are briefly discussed and then an analysis of the effectiveness of the system is presented. This covers the ratio of positive to false detection of the gait. Chapter nine also recommends training the mice for use with the device to acquire higher quality gait data.

Chapter ten concludes the thesis by analyzing the gait data acquired and discussing the quantitative measures derived from the data. This leads into a discussion of the effectiveness of the system to replace or augment the present method of measuring gait recovery. Chapter ten also includes a discussion of what future refinements may be made to the system to possibly achieve improved results. Lastly a final review of the complete system is presented to conclude the thesis.

2 SURVEY OF TECHNOLOGIES

2.1 Introduction

The ideal gait acquisition design solution is to have an arbitrarily long pad on which any small animal can run across that measures the force and placement of each footfall. This chapter will present a variety of technologies which can be adapted to detect both force and position, some of which are suitable for use in a gait sensing device. Additionally, products already on the market that have functionality similar to the proposed system or may be adapted for use in this system will be reviewed. The chapter concludes with a discussion of which technologies merit further experimentation and which are determined to not be suitable to the application of gait detection.

2.2 **Resistive Touch Sensors**

Resistive touch sensors report touch as voltage change relative to a base voltage as the resistance of the probe is altered due to an applied force. The method of how this resistance is generated depends on what type of resistive technology is used. The electrical equivalent of the sensor is a potentiometer. There are several different types of resistance sensors, some lending more to force measuring applications, others to positional applications, while others try to have a blend of both.

The first type is the *layered film* resistive sensor. These sensors generate resistance values based on the contact between two conductive films held apart by a nonconductive separator and are typically used in touch screens [11]. Force is sensed by applying a base voltage to one film and probing the voltage at each point on the opposing film. Of this type, there are two subcategories: 4-wire and 5-wire. The 4-wire system employs two conductive circuit films typically separated by nonconductive micro-dots spaced at the desired resolution and overlain on a glass plate. For this setup there needs to be a separate overlay for both x and y measurements. In the 5-wire setup, one of the circuit films is deposited directly onto the

glass. This allows for both x and y measurements to be taken from one layer. It also greatly improves durability as the base voltage is applied to the rigid glass layer which is not being deformed by continuous touches as in the 4-wire setup.

The second type of resistive sensor is a *semi-conductive polymer film* sensor. This type of sensor uses a polymer film which exhibits a decrease in resistance as force is applied [12]. A micro-dot grid of this polymer film is then laid out to determine position; these are typically used in touch pad pointers. Due to the nature of the technology, resistive sensors are capable of sensing forces applied from a finger, a stylus, or any other object. They also allow for very precise force measurements, though commercially available systems typically set the threshold activation force at approximately 30g (this is typically where linearity begins). Smaller forces may be measured with customized hardware geared toward detecting more minute changes in voltage.

A drawback to resistive sensors is size versus resolution. As the overall sensor area increases, resolution decreases. A pad of an area of 6.35cm x 6.35cm would have a sensor resolution of about 4 sensors per mm. If these sensors are to be utilized the gait acquisition system, many small pads of a high resolution would be needed to cover the overall area required. This would mean more custom hardware to address each sensor. Resistive sensors have a durability of ~35 million touches per point. The sensors are entirely resistant to fluids, both in physical design and electrical characteristics.

2.3 Capacitive Touch Sensors

Capacitive touch sensors report touch based on a capacitance change corresponding to an applied force. There are two main types of capacitive sensors. The first type employs two conductive plates which are separated by a dielectric [13]. When force is applied, the two plates are brought closer together and the capacitance changes. This is the technology that current large scale force measurement pads are based on. It is more susceptible to disturbances due to fluids but can be protected to minimize this. It has a durability of about 100 million touches per point. Its force sensing range is the highest of all the sensor types

with the ability to discern forces from as little as 0.1g to 20kg, though not with the same device. The low end range falls from about 0.01g to 500g with a resolution of 0.01g. The spatial resolution is only about 1 sensor per 2mm but that remains consistent for any size area. This type of sensor is also able to detect any object.

The second type of capacitive sensor works by utilizing a wire grid to detect the presence of a nearby electromagnetic field [14]. When an external field is near enough to distort the field generated by the grid, a measurable change in capacitance occurs. This is how most conventional touch pads work. While highly accurate for determining position, force would have to be determined based on how close the object is to the sensor grid while pressing on a material with a known displacement for a given force. Another obvious drawback is that the measured object must generate an electromagnetic field. A finger would be detected, but a fingernail, stylus, or any other inanimate object would not. This technology gives the highest positional accuracy but the lowest force accuracy. It has the spatial resolution of about 20 sensors per mm but overall area can only be a maximum of about 8cm x 8cm. Force sensing, due to being so difficult to setup, is usually only an on/off value with an activation threshold around 50g. This type of sensor is very susceptible to fluids as they tend to distort the fields and affect the measurements.

2.4 Acoustic Touch Sensors

Acoustic sensors send ultrasonic waves through a glass substrate and measure the location and force of wave distortion caused by a touch [15]. Unlike the other technologies, this sensor has no film overlays or printed circuit contact surfaces. This greatly improves the durability. The system uses an elaborate controller to generate acoustic pulses in the glass and measure where there is deflection. This makes it highly accurate, on the order of 15 sensor points per millimeter. The drawback is that it can only detect one point on the surface. Multiple touch points are resolved to the point nearest the wave emitter. While the sensor is capable of detecting force values, its activation threshold is set between 50g and 80g. With the complexity of the controller needed to use the sensor, it would be difficult to manually adjust the force sensing values or read them directly.

2.5 Fiber Optic Touch Sensors

Fiber optic sensors measure the change in the scattering pattern of light traveling through elastomer cells as they are deformed by a touch [16]. They have a force detection range from less than 0.1g to 300g and a positional accuracy of 1 sensor per millimeter. The durability of the sensor array itself is very low, but the surface can be covered with a variety of materials to guard against fluids, scrapes, and tears. The sensor can be made to a maximum width of 30cm and length of 900cm. It has a maximum sampling rate of 8kHz depending on the number of sensor nodes. The sensor is triggered by any type of object. At the time of the initial research, this type of sensor was out of production while the company producing it, Tactex Controls, underwent redesign and refinancing. It is now available again for custom orders.

2.6 Piezoelectric Polymer Film

Piezoelectric polymer film sensors work by taking advantage of the piezoelectric effect. Piezoelectricity is a material property where an electric potential is produced proportional to the amount of strain. This property is unique in that it is also reversible. An applied voltage will cause the material to flex a proportional amount. This duality is also sign dependent, meaning that a positive flex corresponds to a positive voltage, and likewise, a negative flex to a negative voltage. Piezoelectric polymer film sensors are produced by laminating a layer of a piezoelectric polymer, such as Polyvinylidene Fluoride (PVDF), between two conductive traces, typically silver [17]. Commercially available piezoelectric film sensors are produced with a 28µm thick polymer layer and can generate a CMOS level voltage without amplification [18]. By measuring the voltage, the sensor reads as a vibration sensor, that is, only transient forces are measured. Experimentally, forces as small as 0.5g and as large as 50g have been detected [19]. An interesting property of piezoelectric polymers is that the resonance frequency of the material is dependent on the strain placed on the material. Therefore a steady-state force can be measured as a proportional change in the frequency response of the material as shown in Equation 2.1.

$$\frac{f - f_0}{f_0} = AF \tag{2.1}$$

where f_0 is the natural resonance frequency; f is the resonance frequency after pressure is applied; A is a proportionality constant; F is the applied force;

2.7 Machine Vision Cameras

Machine vision cameras differ from conventional digital PC cameras in that they typically offer greater resolution, frame rate, feature sets, and configuration options. The basic requirement of a machine vision camera is that it must meet higher resolution and frame rate demands. Typically, machine vision cameras have variable resolution and frame rate setting that are maximized based on available bandwidth. This can mean real-time capture (30fps) at high resolutions (1280x1024) or high speed capture (100+ fps) at smaller resolutions (640x480). Beyond those two settings, focus adjustment, wide angle viewing, and a PC connection are vital camera features. Focus adjustment allows the camera to be placed varying distances away from the object and still maintain image clarity. The wide angle viewing area allows for a shorter "working distance" or distance between the camera and the target. This is important for maintaining a minimum device height to ensure tabletop usage (~6"-12") while still allowing a greater span (~8"-24") to remain in view. A PC connection (Firewire or USB) is also preferred as the data is to be stored and analyzed digitally.

Converting from an analog to digital using a frame grabber is inefficient, requires more hardware, and has the possibility of introducing noise. Additional camera features include control, digital or otherwise, over gain, contrast, and exposure time. Variable gain allows the camera to adapt to a variety of lighting conditions. A feature which will become important when numerous lighting schemes are tested. Brightness and/or contrast controls allow the intensities in the image to be fine tuned to help highlight relevant data. Control over exposure time helps to reduce motion blur by capturing over a very short time span, thereby adding clarity to the data. Further detail on machine vision cameras will be discussed in following chapters.

2.8 Kistler Force Plate

The Kistler Force Plate is a smooth surface connected to four 3-axis piezoelectric force sensors. The three axes allow forces to be measure in all three directions. This product is used for gait experiments where component force measurements are needed. Measuring ground-reaction forces (z-direction) only shows how much mass is being supported by each limb. Measuring forces in the x and y directions shows how much force is being applied to forward locomotion or braking. With the addition of specialized hardware, this system is also able to determine the position where the force is applied. The system can only resolve a single point of contact however and is intended for measuring human gait. The plate dimensions are 24cm x 40cm and it has a sensing range from 0kg - 510kg.

2.9 Fingerworks Touchpad

The Fingerworks Multipoint Touchpad is designed for low impact mouse and keyboard input. Its based on a custom designed capacitance sensor array sampled quickly allowing it to track multiple points simultaneously. Commercially it is intended to be used as an alternative computer input device which allows hand gestures to be translated to keyboard commands. The raw data stream from the device can be accessed directly to provide position and surface pressure of all points of contact. The pad itself is 6.5" wide by 5" long and has a sensing array of 40 x 16 elements, resulting in a resolution of three sensors per inch on the vertical and six sensors per inch on the horizontal. Tests with live mice show an adequate

response, though the spatial resolution was too low to obtain data other than crude paw placement. This is shown in Figure 1. A software development kit was to be released for the device, however, from the time the initial research was conducted, Fingerworks has since gone out of business and this product is no longer available.



Figure 1. Live Animal Response of Fingerworks Touchpad

2.10 Pressure Profile Systems TactArray

The TactArray by Pressure Profile Systems (PPS) is a thin, flexible, sheet that can sense force and position via a high precision measurement of a capacitive sensor grid. The sensor grid is created by layering vertical and horizontal conductive strips separated by a compressible dielectric. The capacitance of each intersection is measured to determine force applied. The system can sample a maximum number of 10,240 elements by measuring ten 32 x 32 element pads simultaneously. The entire array can be sampled every tenth of a second. Each sensing element is 4mm². Forces as low as 1g are able to be detected. The TactArray is only available as a complete system, custom designed for each customer. The sensing pads themselves are not sold individually .

2.11 Tekscan Animal Gait Measurement System

The Tekscan Animal Gait Measurement System provides a full hardware and software solution for the measurement of small animal gait. The sensors are built from a combination of force sensing resistors and piezoelectric film arrays [20]. The sensors are customizable in size and resolution from a minimum size of 5.6cm x 5.6cm and a resolution of 1.3mm per sensor to a maximum size of 24.6cm x 98.4cm and resolution of 5.6mm per sensor. The ideal sensor for detecting mouse gait would be the model 5051 with a size of 5.6cm x 22.4cm (4.4"x8.8") and a resolution of 1.3mm (0.05") per sensor.

2.12 Mouse Specifics DigiGait Imaging System

The Mouse Specifics DigiGait Imaging System is a full gait imaging and analysis device [21]. It uses a transparent belt treadmill for the animal contact area and an underside mounted camera to capture the gait data. Interchangeable compartments are placed above the belt to contain the animal and are provided for both mice and rats. The treadmill belt speed is variable from 0 to 100 cm/s and can be set to an up or down incline. Gait detection is done semi-automatically by matching based on paw color and a user selected contrast ratio. The maximum capture frame rate is 150 fps. Twenty-five different gait metrics are reported. These are based on spatial and temporal results which include step sequences and averages of total and single paw stride distances, contact durations reported in times and percentages, and paw area and rotations. The system is not available for purchase, only for rent from the company. Figure 2 shows a sample screen displaying gait metrics acquired by the system. Average stride length, paw angle, stance width, and timing data can be seen.



Figure 2. DigiGait Analysis Screen

2.13 Discussion

The initial design solution was to have an arbitrarily long pad on which any small animal could run across and the force, placement, and timing of each footfall be measured. A variety of technologies were surveyed which could be adapted for use in a gait sensing device. After thorough examination of the available technologies, it was found that many could detect position or force, but few could do both. It was also found that many of the position sensing technologies could only resolve a single contact point and were therefore unsuitable for gait detection. Additionally, there were several products already available that had similar functionality to the proposed system. Each of these, however, were found to be too costly, lacked a key feature, or were not adaptable to the requirements for use in the gait detection system.

Experiments with the resistive based sensors showed several limitations of the technology. It was determined that the sensors were not sensitive enough to measure the small forces applied by a mouse. The sensors were also incapable of resolving multiple contact points, the result being that a sensor array would need to be constructed to acquire positional data. It was found that there was no readily available manufacturing process to create an array of resistive sensors. For these reasons, the resistive based sensor was dismissed. The acoustic sensors were unable to locate multiple touch points as well and were overly complex to setup

and maintain, and therefore were not feasible. The fiber-optic sensors could have been an acceptable solution, however, they were not available for purchase or testing during the development phase. The Kistler Force Plate would have measured all three force vectors very accurately, but lacked any kind of positional data without complex analysis hardware, and even then could only resolve a single point of contact. The PPS TactArray and the Tekscan Gait Measurement System would have been ideal solutions for preliminary development, but only complete, custom systems were available, and these did not meet all the requirements of the proposed system. Individual components which could be modified and adapted were not available for purchase. The Mouse Specifics DigiGait treadmill is the device with the most similarities to the final system. It was, however, not available for purchase, only for rent, and therefore not customizable.

The capacitive based sensors, the piezoelectric film sensors, and the machine vision cameras were the three technology types which showed the most possibility for use in gait tracking. Capacitive based sensors can be grouped closely together, thus creating high resolution arrays and allowing for multiple contact point detection. They also have the ability to detect force based on proximity or contact area. Piezoelectric film sensors can also be configured in high resolution arrays which would permit multiple contact points. They too have the ability to detect forces, either transient from direct measurement, or static from material strain measurement. Lastly, machine vision cameras provide the best spatial resolution of all technologies as well as providing a easy interface for data capture. The disadvantage is that force data is lost. These three technologies proved to have the most potential applicability to the design of a gait detection system. Further research was therefore conducted to determine the merits of each type of technology by building actual hardware prototypes and testing with live animals.

3 TECHNOLOGY FEASIBILITY EXPERIMENTS

3.1 Introduction

Three of the technologies surveyed, capacitive sensor arrays, piezoelectric film sensor arrays, and machine vision cameras, merit further experimentation to determine if they can be applied to gait detection. This chapter will begin by presenting a brief discussion of addressing schemes for large sensor arrays. This is necessary to understand the complexity involved in producing the capacitive and piezoelectric film sensor arrays. Next, a dual layer piezoelectric film sensor device will be presented. Following that, a capacitive grid sensor device will be discussed. Finally, a system designed using a machine vision camera will be explored. The chapter will be concluded with a comparison of the features and limitations of each of the technologies based on actual experiments.

3.2 Addressing Schemes

To help understand the prototype system designs, it is necessary to present several methods of addressing arrays of sensor elements. There are three primary issue to discuss when selecting an appropriate addressing scheme: resolution, cross-talk, and wiring complexity. Resolution relates to the density of the sensor nodes. For sensing mouse paws, it has already been stated that resolution must fall within 10 nodes per inch to 100+ nodes per inch to obtain relevant data. The individual sensor nodes, therefore, must measure, at the very most, no larger than one tenth of an inch. Additionally, there must be a buffer space between the sensor elements to isolate them from one another and to reduce cross-talk. Cross-talk occurs when adjacent sensor elements not in contact with the object report a value due to their proximity to the activated sensor. High sensor resolution becomes irrelevant if the cross-talk between all the sensor nodes is too high. This could allow hundreds of small sensors to report instead as a single large one. With high resolution and small, numerous sensor elements, wiring complexity increases. A single square inch of sensor area with a resolution of 10 nodes per inch can have 100 output lines. The same area with 100 nodes per inch can

have 10,000 output lines. A 3" by 12" sensor pad at high resolution can have as many at 360,000 output lines. Direct addressing of each sensor element at this size and resolution is prohibitive due to the number of circuit traces which must be made, the physical layout space required, and the volume of circuit components to which each trace must be connected. It is for these reasons that more novel addressing schemes are explored. There are three types of addressing configurations other than direct addressing which were developed and evaluated: dual layer busing, passive element matrices, and active element matrices.



Traces go to backside to avoid intersecting.

Figure 3. Dual Layer Bus Addressing

In the dual layer bus addressing scheme, shown in Figure 3, the circuit traces act as the sensing surfaces for electromagnetic field detection or capacitance changes. This uses a dual layer circuit board with the sensing columns on one side of the board and the rows on the other. The signal gain for the underside of the board would then be adjusted to match with the top side which is closer to the point of contact. An alternate design is to have both traces on a single side of the board with the rows being shunted to the back of the board only when

they would intersect a column. This also reduces the gain matching issues. A single, nonconductive layer is placed over the traces to protect them from direct contact. Sensing is done by row and column. The disadvantage of this method is that with multiple contact points, detections will occur at any active row-column crossing. Two contact points, in opposite corners, will be falsely identified as four contact points arranged as corners of a rectangle.

The second type of addressing scheme is the passive element matrix design. In this design, shown in Figure 4, the sensing elements are unpowered circuit elements, such as force sensing resistors or capacitive plates, which are connected to multiplexed horizontal and vertical bus lines. The bus lines are used to complete a circuit through an individual sensor element and then the circuit properties are measured. When not addressed, the bus line goes to a disconnect state. Each element is then switched into the circuit, one by one, until the entire grid is read. The advantage is that the number of output wires are reduced from *rows* * *columns* to *rows* + *columns*. The disadvantage is that the nodes which are not being actively addressed are electrically summed and then placed in parallel with the addressed node, thus adding extra cross-talk to the signals.



Figure 4. Passive Element Matrix Addressing

The third type of addressing scheme it the active element matrix design. In this design, shown in Figure 5, the sensing elements are powered nodes which are connected to power and ground buses and have a third output line which carries the sensor data. Each element is selected by powering connecting it to power and then reading the sensor value from the output. The entire matrix is read by cycling power to all nodes. For some sensor types, such as piezoelectric sensors, the output bus can be reduced to a single wire connected to the output line of all the sensor elements. This has the possibility of increasing sensor cross-talk however. An alternative is to only connect the outputs from each row or each diagonal set of sensors. This has the benefit of reducing output line numbers with minimal increase in sensor cross-talk.



Figure 5. Active Element Matrix Addressing
3.3 Dual Layer Piezoelectric Film Array

This sensor array prototype is comprised of two piezoelectric film layers separated by a thin, non-conducting layer. The bottom piezoelectric layer is excited by a high frequency signal. This causes the layer to vibrate. The vibration is then transmitted mechanically to the top layer were it is measured. When a force is applied to the surface, the vibration is dampened. Force can be measured as a proportional decrease in the amplitude of the signal. A simple prototype of this design was built and tested using piezoelectric film sensors from Measurement Specialties, Inc. (MSI). The sensors are shown in Figure 6 and the sensor array prototype is shown in Figure 7.



Figure 6. Piezoelectric Film Sensor



Figure 7. Dual Layer Piezoelectric Touch Sensor Test

This prototype consists of three dual layer piezoelectric pads placed adjacent to one another. These are the three strips located in the center of the breadboard in Figure 7. A frequency of 44kHz was applied to the bottom film layer and the top layer of each pad was individually monitored on a four channel oscilloscope. It was shown that when a force is applied to a pad, the corresponding signal amplitude decreases. This is shown in Figure 8 where the signal of the sensor being touched has decreased approximately 0.5V.



Figure 8. Response of Dual Layer Piezoelectric Sensor to Touch

Further testing showed that there are several more factors to be taken into account to make the system perform reliably. It was found that an intermediate layer must be sandwiched between the two piezoelectric layers. In addition to being non-conductive, this layer must also be thick, relative to the piezoelectric layers, and deformable. This broadens the measured voltage range of the force dampened signal. It was also found that much of the signal response was not due to the pressure applied, but rather to the capacitance storage created by the two conductive films. It was difficult to measure amplitude changes using a small mass set but was very easy when contacting the sensors with a finger. The natural EMfield surrounding a finger was enough to distort, to a measurable degree, the capacitance created by the layered sensors. Although this was proportional to applied pressure, it was also very easy to force the value to its extreme by pressing the sensor stiff against the breadboard or by increasing the area in contact with the sensor. Due to these design challenges, it was decided to proceed with a different sensor type rather than try to create and characterize a reliable piezoelectric solution and then adapt it to the needs of this project.

It should be noted that a similar design was presented in the October 2004 issue of the IEEE Sensors Journal [19]. The research presented uses an array of single layer piezoelectric elements to measure an applied steady-state force. This method is similar to the dual layer piezoelectric force sensor in that it stimulates a sensing element with fixed frequency signal (in this case, the natural resonance frequency) and then measures the amount of signal change when a force is applied. The difference is that this method both stimulates and measures from the same single layer of piezoelectric material by taking advantage of the property of the material to shift its natural resonance frequency based on applied strain. The results of this method show merit, however, considerably more research would need to be done before this method of sensing could be employed in the design of a gait detection system.

3.4 Capacitive Sensor Array

The capacitive sensor array measures a very small change in capacitance of a conductive plate when a external EM-field, such as that generated by a finger or a paw comes in proximity. An array of plates allow both force and location to be determined. The Fingerworks Touchpad utilizes this type of sensor technology. Beyond the initial tests to determine if a mouse would be capable of activating it, the Fingerworks Touchpad was not further experimented with. This was due to the company's lack of development kits and then its eventual closing of business. A capacitive based sensor prototype was therefore developed using Quantum Research Group's capacitive touch sensor chips.

The Quantum Research Group's QTouch, and QMatrix lines of chips provide single chip solutions for creating and configuring capacitive sensors and sensor arrays. The first test design used the QT110 chip, which is a single input proximity sensor, shown on the left in Figure 8. This came mounted on an evaluation board directly from the company and was used explore the potential of the capacitive sensor based solution. The second test design used the QT320 chip, which is a dual input proximity sensor, shown on the right in Figure 9.



Figure 9. QProx Single and Dual Input QTouch Boards

Both of these chips were set to trip the circuit when an external field was present within 2mm of the sensor surface, or the thickness of the printed circuit board. The sensing surfaces were traced on the underside of the board and the active sensing area was marked on the top. The dual input chip was used to drive a rudimentary proximity sensor array. The sensor array was designed using the dual layer addressing scheme. The rows and columns were multiplexed using a Maxim MAX306CPI dual 8-bit analog multiplexor. This initial design, though not very effective, did highlight the false detection issue with the dual layer bus addressing structure. It also demonstrated the inability to multiplex effectively with capacitive sensors which auto-adjust.

The third test design used the Quantum QT60645 chip, a 64-key proximity sensor, mounted on the E6645 evaluation board. In addition, the evaluation kit came with a 64-key keyboard layout, and control software. The evaluation board, sample keyboard layout, and a sensor array built for testing are shown in Figure 10. The control software is used to adjust circuit gains and to monitor contact points. Testing was done by first placing a mouse on the keyboard layout to determine if the mouse was capable of registering contact with the system. It was found that it was.



Figure 10. QProx E6645 Evaluation Kit

The next step was to create a custom grid using the same 64-key input but with a very small sensor spacing. A circuit board layout was drawn to mount the QT60486 chip; a 48-key chip. Quantum did not offer low volume orders or samples for the QT60645 chips and has since ceased producing them altogether. The board layout was designed to connect only the minimal set of required components and provide a connection to the sensor grid. The sensor grid itself was designed as a second circuit to be printed on a flexible film. The grid is composed of interdigitating wire traces separated by ground buses. The circuit diagram for the sensor array is shown in Figure 11 and the dual side film layout for the sensor grid is

shown in Figure 12. Sensing works by connecting row and column traces to the chip, which then measures the baseline capacitance between each interdigitated trace. A deflection past a set threshold of the capacitance field by an external field indicates a touch. The ground traces are used to separate one sensing element from another by absorbing stray capacitance. The sensor grid would be mounted to a rigid base and then connected to the drive circuit, mounted underneath. This allows for multiple individual grids to be placed adjacently, thereby creating a much larger, unified sensing area. A master control processor would then be used to query each individual sensor grid and collect and report data from the entire field.



Figure 11. QT60486 Sensor Chip Circuit Diagram



Figure 12. Interdigitated Capacitive Sensor Grid Layout

The capacitive sensor array had several production limitations which led to the ultimate decision to abandon the design. Firstly, it was deemed too costly to produce the micro-trace, flexible film, sensor grid. Local facilities were not available to manufacture the grid so it had to be made by an outside board manufacturer which specialized in thin film layouts. Small, precision, low quantity units proved to be too expensive to produce and the turn-around time proved to be too long. These limitations were compounded by need to test, redesign, and then manufacture several iterations. In addition to prototype production limitations, the sheer number of grids which must be connected together to form a complete sensing field became a concern. Each grid was only sized to be 1"x1", which would require 36 units to achieve the basic size of 2"x12" or 72 units to achieve an optimal size of 3"x24". It also proved difficult to design a circuit layout compact enough to fit all the necessary components into a single square inch area. Even if the thin film sensor grid and the circuit layout could have been efficiently produced, the sensor grids would only have a resolution of eight sensing elements per inch, too coarse of a resolution to obtain any data other than basic contact points.

3.5 Camera/Plate System

The camera/plate system is based on a machine vision camera, mounted under a transparent or translucent plate, which records the gait of a mouse as it walks across the plate. The recording is then analyzed to extract the locations of each paw placement. Several factors were considered during the selection of an appropriate camera. It was decided that the camera needed to have a digital output or an external frame grabber for transferring the video to a computer; necessary for both analysis and archival of the video. Other important factors include resolution, frame rate, feature sets, and data connection type. Table 1 shows a comparison of the various cameras which were considered.

					Max			Internal
		Frame	Adjustable	Adjustable	Shutter	Mono /	Connection	Frame
Camera	Resolution	Rate	Resolution	Frame Rate	Speed	Color	Туре	Grabber
							BNC	
							Connector /	
Pulnix TM6703	640x480	60fps	No	Yes	1/32,000	Mono	12-pin Video	No
Pixelink PL-A741	1280x1024	27fps	Yes	Yes	1/25,000	Mono	Firewire	Yes
	1000x1000	34fps						
	640x480	104fps						
Basler A301f	640x480	80fps	No	No	-	Mono	Firewire	Yes
Sony XC-HR50	640x480	60fps	No	No	1/100,000	Mono	12-pin Video	No
Imperx IPX1M48	1000x1000	48fps	Yes	Yes	1/16,000	Both	12-pin Video	Yes
	640x480	80fps						
Lumenera LU070	640x480	60fps	Yes	Yes	-	Both	USB 2.0	Yes

Table 1. Machine Vison Camera Comparison

The cameras were compared primarily on maximum resolution and frame rate. These two factors directly correlate to accuracy of the spatial and temporal gait measurements. The connection type was also important since it dictated if special hardware was needed to get the data from the camera. Additional features, such as adjustable resolution and frame rates, were considered too, as they allow for more flexibility of experimentation.

Although high-end machine vision cameras had been researched, the first camera experimented with was the Logitech Quickcam for Notebooks, shown in Figure 13. It was decided to begin with a basic consumer digital camera to test the feasibility of the

camera/plate design before purchasing an expensive high-end camera. Although the Quickcam did not have the larger resolutions, faster frame rates, and extensive feature sets of many of the machine vision cameras, it was inexpensive and easy to use for initial testing. The Quickcam has a frame rate of 15fps at a resolution of 640x480 and 30fps at a resolution of 320x240. It has a manual focus lens and feature controls which include adjustments for digital gain, gamma correction, and white balance. It can run in either color or grayscale mode at both resolutions and it connects to the computer via the USB port. It is a Video for Windows compatible device, so nearly every Windows based video capture program will recognize it. The Quickcam was used with the Pagoda prototype device, detailed in a later section.



Figure 13. Logitech Quickcam PC Camera for Notebooks

3.6 Discussion

The goal of this research is to build a sensing device to measure the force, position, and timing parameters of the gait of a mouse over several strides. Data would be collected by a sensor array, preferable a solid-state sensor device embedded within the walkway, and would be communicated, to a central controller or user console. After an extensive survey of sensor devices, the three most promising technologies underwent experimental testing.

The piezoelectric film sensors have a number desirable characteristics including compact size, static and transient force sensitivity, and simple circuit connectivity. An array of densely packed piezoelectric film sensors would provide data on both the location and the applied force of multiple, simultaneous contact points. The limitation of the technology is found in the inability to easily manufacture such a sensor device. The basic testing conducted

with only three sensor elements demonstrated the difficultly in creating and calibrating a piezoelectric film sensor array. The final conclusion is that such an array must first be thoroughly researched to determine the optimal film thickness, piezoelectric strain, intermediate layer material, and capacitance rating, among other factors before any manufacturing can be done. The production process itself must then include screen printing or deposition of the piezoelectric film, non-conductive layer (if applicable), and all addressing lines with very small, precise tolerances. Such a research effort would be large scale project of its own. A manufacturing process would also have to be developed to create the final sensor device. The piezoelectric thin-film sensor array goes far beyond the scope of this project which is why it was abandoned for an alternate sensor technology.

Capacitive sensor arrays have the advantage of not having actual sensor elements, but rather function by measuring the capacitive field between two wire traces. This allows for a very simplistic circuit design since no additional components need to be connected beyond the single chip which measures the capacitances. It also has the advantage of allowing for a high resolution sensor grid. Additionally, the sensor chips used in the test design automatically calibrated to the sensor grid. This prevents the sensitivity from straying over time or with use. The experimentation with this technology progressed to the design of a full sensor array prototype. It was at this time that the inherent disadvantages of the capacitive sensor array solution became apparent. The wire traces and spacing between can be made to be very small (down to 0.005") on a printed circuit board or flexible film. At the finest trace size, a full sensor element, comprised of a ground wire, a column wire, a row wire, a second ground wire, and the corresponding space between each of them, will measure 0.04"x0.04". At that size, the resolution is 25 sensors per inch, or roughly 1 sensor per millimeter. This does not include any addressing line which must be added. This design assumes a single wire for each row and column with no interdigitation. When setup in this configuration, the signal gain for each sensing line must be increased since there is so little capacitance to measure. This has the disadvantage of drastically increasing the noise level of the signal which causes an increase in false contact detections. In practice, it was found that the chip could not calibrate properly and the sensors were either forced constantly active or inactive depending on where

the threshold was set. To create a grid which was not overly amplified, the resolution was reduced down to only eight sensor elements per inch. The second major limitation of the capacitive sensor array was the same problem with manufacturing that was found when exploring the piezoelectric film approach. The row and column sensor grid traces are required to be on or very near to the same plane. This poses a layout problem when printing the grid on standard PC boards. It was therefore required to print the grid on a flexible film with the vertical and horizontal traces on opposite sides. Again, manufacturing became an issue due to the expense involved in producing thin trace, high precision, low quantity, custom circuits. It was ultimately decided to not use the capacitive sensor array technology based on the aforementioned limitations.

The machine vision camera has the distinct advantages of a high spatial resolution, fast frame captures, and an easy interface for data collection. The two clear limitations are that the capture area does not scale well and that there is no way to collect force data. The spatial resolution is dependent on three factors, the overall camera resolution, the distance to the target, and the viewing angle of the lens. With the right lens and the proper height adjustment, a high-end camera can cover a full 12" span at a resolution of 100 pixels (sensors) per inch. The proper camera can also sample this area at a rate of 100 frames per seconds, a difficult feat with large solid-state sensor arrays. Additionally, the camera does not require specialized hardware to be designed to collect and transmit the sensor data. All but one of the machine vision cameras come with an internal frame grabber and connect via standard digital connectors. The limitation of the camera design is the inability to measure applied force. After reviewing the first two sensor technologies, and the metrics actually required to measure gait, it was determined that removing the force measurement in exchange for the spatial accuracy of the camera was an acceptable tradeoff. It was therefore decided to proceed with the machine vision camera based design. This was then adapted to the camera/plate system where a rigid, transparent plate would be mounted above the camera. The camera then records video of the mouse as it runs across the plate. This simplified system of data acquisition would then allow the research to be focused back to gait identification rather than sensor development.

4 GAIT ACQUISITION AND ANALYSIS DEVICE SPECIFICATION

4.1 Introduction

The following chapters will detail the development of a camera/plate based gait acquisition system and an image processing based detection and analysis tool. Chapter Five will detail the specific machine vision camera which was chosen for the final implementation. Chapter Six will focus on the design evolution of the physical walkway stand assembly. Several design challenges were discovered during the development process. Discussion will cover the importance of each to the overall design and the progression of a solution through each prototype iteration. The final walkway stand design will be presented which incorporates the refined hardware features and meets all the system requirements. Chapter Seven will begin the discussion of the automated gait detection algorithm by presenting the many image processing techniques explored. Chapter Eight details the two image processing based paw identification algorithms used and describes the final implementation. The end result of this work will be a complete gait acquisition and analysis system consisting of a well designed physical stand with considerations for an optimal data acquisition environment, animal management, and simple user operability, and a robust paw identification algorithm based on advanced adaptable image processing methodologies.

4.2 System Requirements

The system must be able to capture video frames from a minimum frame rate of 60fps to an ideal rate of 100fps. Experiments conducted with live mice using a 60fps camera show paw contact and swing durations as low as four frames each. This is the minimum number of frames acceptable per strike. Using a lower frame rate could result in the complete loss of a strike if the mouse is moving at top speed. The capture area must have a minimum length long enough to acquire three complete strides of a mouse. The longest stride length measured from the live experiments was 6.4cm or 2.5". A walkway length of at least 8" is therefore required; 12" would be ideal. The image resolution must meet a minimum of 100 pixels/inch

to properly discern the digits of each paw. This was determined by capturing the image of a paw at 100pix/in and then reducing the image size and evaluating the clarity of the digits. Digit spacing is the most discernable paw feature and will become the basis of the detection algorithm, it is therefore important to have the greatest amount of resolution possible. The walkway stand must be small enough for table top use, lightweight enough for easy movement, and robust to repeated handling. The stand must be self-contained, therefore, all the necessary sensors, electronics, lighting, and the camera must be housed within the stand. The mouse must be able to be easily loaded into the device and have a directed path over the active viewing area. To accommodate the mouse, the walkway must be a minimum of 2" wide; 3" to fully accommodate an injured animal. The walkway must be tolerant to repeat mouse movement across the surface and robust to any excretions the mouse may leave behind. It must also be easy for the operator to clean and to retrieve the mouse at any point. The device must maintain set lighting conditions at all times to ensure uniformity in the image data. However, the algorithm must be tolerant to non-uniformities in the lighting should optimal lighting not be guaranteed. The algorithm must execute in a timely manner and display results to the user in a meaningful and effective way. Lastly, the algorithm must be able to automatically detect and identify only the paws in a frame which are in contact with the plate and then report a minimal gait parameter set: this includes step sequence, stride length, stance width, stride time, and paw rotation.

4.3 Design Overview

The camera/plate system design will consist of a machine vision camera mounted beneath a transparent/translucent plate which will serve as a walkway for the mouse. The machine vision camera will be the PixeLINK PL-A741-R. The plate will be constructed out of a glass or plastic pane. An enclosure will be constructed over the walkway to direct the mouse across the active capture area and to assist with the loading and retrieval of the animal. Trip sensors will be placed along the sides of the walkway to notify the system when the mouse has entered and exited the capture area. Notifying the system of when an animal is present will trigger data capture and thereby limit the video to only the frames which contain the animal and are therefore relevant. Lighting will be mounted to the stand to properly

illuminate the walkway for improved data capture. An optimal lighting scheme is important to produce the maximum contrast possible between the paws touching the plate and the rest of the mouse's body and the background. This will help the detection algorithm to separate paws in contact with the plate from the rest of the scene. Additional hardware will be in the form of analog/digital circuitry and/or a microcontroller to handle sensor interfacing and communications to the host PC. A host PC will be used to run the software designed for capturing and storing gait data from the camera. Additionally, the paw identification and gait analysis algorithm will run from the host PC. The identification algorithm will utilize image processing methods to extract and identify the individual paws in the frame. This will be done in a three step process: locating relevant data, categorizing the data, and applying object matching techniques. Once located, the positions of the paws will be used to calculate all the spatial gait parameters and the frame number combined with the capture frame rate will be used to calculate all the temporal gait parameters. The spatial and temporal accuracy of the gait measurement will be dependent then on the resolution and frame rate of the camera.

5 MACHINE VISION CAMERA

5.1 Introduction

The PixeLINK PL-A741-R camera, shown in Figure 14, is a professional grade machine vision camera. It supports a number of features and controls and is highly reconfigurable for adaptation to a variety of imaging applications [22]. It is a grayscale camera and connects via a standard Firewire port. It supports a standard C-mount, 2/3" CCD lens. It has full software control over all features, such as gain, brightness, exposure time, resolution, and frame rate. It supports external hardware frame triggering and has two general purpose output pins. The camera is setup in a right angle configuration for easy mounting into spaces where height is a concern.



Figure 14. PixeLINK PL-A741-R Machine Vision Camera

5.2 Capabilities

The camera's key feature is its ability to provide a variable resolution and frame rate with the limiting factor being maximum data transfer bandwidth. The camera's resolution can be adjusted from a minimum of 80x2 pixels to a maximum of 1280x1024 pixels or to any user defined value in between. Additionally, the capture area can be windowed to any region within the boundary of the maximum resolution. For example, the capture area can be set to a 640x480 window with an offset of 300 pixels from the top and left edges. The camera can achieve a maximum frame rate of 8000 fps at the minimum resolution and exposure time, At the maximum resolution and exposure time (40 milliseconds), the maximum frame rate available is 13 fps. The resolution and frame rate are dependent on exposure time and available bandwidth. The camera's total available bandwidth is ~280Mbps. The exposure time defines how long the camera integrates light in the CCD before storing the value and resetting the sensor. Decreasing exposure time allows for a greater number of frames to be captured per second. However, the brightness of the image is reduced since there is less light collected by the sensor. The camera can achieve a maximum frame rate of 27fps at full resolution when set to the minimum exposure time of 0.04ms. The external lighting must be very intense at this setting for the camera to register any image data. A spreadsheet to calculate the camera's frame rate at various resolutions and exposure time settings is provided by PixeLINK.

The original version of the camera supported three shutter trigger modes: rolling shutter, free running global shutter, and hardware triggered global shutter. The rolling shutter mode would integrate the light values in a single row of the CCD, pass the data onto a buffer, and then continue with the next row. This method provided a higher overall frame rate, but resulted in blurring quickly moving objects within a frame. This shutter mode is no longer present in current versions of the PL-A741-R camera and all data pertaining to it should be disregarded. In global shutter mode, the CCD integrates lights over all the sensor rows simultaneously and stores the values in a buffer. When the integration time expires, all the image data is transferred at once, and the sensor resets to gather data for the next frame. In the free running mode, the camera generates an internal periodic signal based on exposure

time and frame rate to trigger the capture of frames. In hardware triggered mode, the camera looks for an externally generated trigger signal on the trigger port pins.

5.3 Software Development Kit

The camera package includes a software development kit (SDK) for the integration of the camera into third-party applications [23]. The SDK provides a software interface for adjusting the camera settings and for acquiring image data. It also provides support for callback functions for on-the-fly image processing. The SDK has two main parts: the OEM Capture Application and the Application Programming Interface (API) Library.

PixeLINK Capture OEM	
View Camera Help	
Video Capture Descriptors External C Basic Controls LUT and FFC	Control Callbacks Extended Shutte Region of Interest Image Capture
Camera Select PixeLINK Camera s/n : 7410314	Video Preview Play Pguse Stop
Egposure Time (ms)	<u>G</u> ain (d8)
0.04 5 Auto 1000	0 13.2 Auto 192
Saturation (%)	Brightness (%)
0 0 <u>Auto</u> 100	-15 0 Auto 15
Gamma [[Dn]	Fiame Rate (Ips) 17 (On)
White Balance Red	Color Temp. (K) [7] [2-1]
Blue 0 Auto 100	Size [1152 x 320
Image Capture File Name: D:\Documents and Settings\ca File Fgmat: AVI	rey/DesktopVFFC7.avi Browse No. of Fjames: 50 Capture
Captuge Full Frame Foreinger Captuge Full Resolution Preview	nt File Name After Capture Image After Capture

Figure 15. Screen Capture of PixeLINK Capture OEM

The OEM Capture Application is a generic application provided by PixeLINK which gives access to all the camera's settings and provides an out-of-the-box means to capture images or video. A screen shot of the OEM Capture Application is shown in Figure 15. The API Library is a compiled external development library containing a collection of API functions. The API functions are medium level routines which provide all the basic commands and control of the camera; from setup parameters to data acquisition.

In addition to the image data, with each frame, the camera sends a frame descriptor. The frame descriptor lists all the frame's parameters, such as image size, frame number, exposure time, as well as the values each of the cameras settings during that frame. The descriptor may be customized to provide additional user-defined data.

5.4 Limitations

Although the PixeLINK camera is highly versatile, it remains limited in several ways. The maximum bandwidth for a Firewire port is 400Mbps. The camera only transmits data at 280Mbps. There is a lot of untapped bandwidth which the camera could use to increase the maximum frame rate at maximum resolution. Also, the camera only captures grayscale images. Color images would help to lessen the demand for high contrast data. Instead of a cutoff threshold being set to distinguish between foreground and background objects, shades of color could be extracted for better processing. A true GPIO port would be very beneficial for interfacing with other external hardware.

The real limitations of the camera are seen in the SDK. The OEM Capture Application is too simplistic to be used for anything but device setup and testing. The API Library is only a loose collection of Windows API functions. The library is not organized into any kind of class or object oriented structure, despite its intended use in a C++ development environment. This means that a wrapper class must first be written to encapsulate the API functions and all associated variables before the camera can be accessed. The functions themselves are not low level enough either to provide complete access to the camera's data registers or other internal hardware. What results are functions which do not provide the

hardware access that assembly routines would, nor the ease of use of higher level, object oriented solutions. These limitations, however, can be designed around and the PixeLINK camera retains best feature set is the most customizable of all the cameras. For these reasons, it was selected for system development.

5.5 Lenses

The camera lens had to be carefully selected to provide greatest area of coverage with the shortest working distance and least distortion possible. To select an appropriate lens, the working distance, or distance between the lens and the walkway, the desired length of the capture area, and the desired resolution of the image had to be determined [24]. These variables would decide the focal length, or distance between the lens and the CCD sensor element, required. It was decided to set the image resolution to 100 pixels per inch as this provided adequate imaging of both the paw's rotation and spread of the digits. Figure 16 shows images of the same paw at successively smaller resolutions, in descending order: 150 pix/in, 100 pix/in, 75 pix/in, 50 pix/in. At the resolution of 50 pix/in, the paw shows individual digits and rotation are difficult to distinguish. Without the side-by-side comparison to the higher resolution images, identification would be even harder. The 75 pix/in paw begins to show some difference in the digits, but it would again be difficult to judge rotation without the comparison to the higher resolutions. At 100 pix/in, the resolution is just high enough to distinguish all the pertinent characteristics.



Figure 16. Comparison of Descending Imaging Resolutions

Based on that resolution, the length of the capture area was set to 12" or 1200 pixels (<1280pix max). To maintain a high frame rate, the capture area was windowed so that only a long narrow portion was active. It was decided that the most width needed would be 4"

since the animal easily fits into that width and it provides ample room for an injured animal to spread out its limbs. A working distance of 20cm or less was also determined for maintaining a small device height. Based upon these values, a focal length of 5mm was calculated using equations 5.1 and 5.2 and Figure 17 [24].



Figure 17. Lens Geometry

$$\tan(\theta/2) = \frac{FOV}{2 \times WD} \tag{5.1}$$

$$FL = h \times \frac{WD}{FOV}$$
(5.2)

For the camera's 2/3" CCD (8.8mm x 6.6mm), this sets the working distance to ~17cm. In application though, a 5mm focal length lens for a 2/3" CCD is rare and is priced far above what is acceptable for this project (>\$1,000). The lens ultimately chosen has an 8mm focal length, with an angular field of view of 57.62°. This allows a 100 pix/in resolution at a working distance of 29.56cm. The lens is a Computar M0814-MP.

6 WALKWAY STAND AND DEVICE HARDWARE

6.1 Introduction

This chapter will discuss the walkway stand and the physical hardware of the gait acquisition device. This chapter also presents the design evolution of the physical setup from a basic test rigging to the final hardware assembly ready for experimental use. All of the attempted hardware designs will be presented in order of implementation to demonstrate what worked, what did not, what led to improved designs. It will also be shown why the elements which made it to the final design are necessary for the overall operation of the device. The chapter is split into three sections. The first discusses all the individual components of the physical device. The second section iterates through four prototype designs, documenting the progression toward a final solution. The last section discusses the various features of the final prototype and how the design requirements were met.

6.2 Components

The following sections detail the core components which comprise the walkway stand. Extensive testing was conducted to find the best hardware for each component type. The technical details of the hardware and which worked the best for each grouping are presented.

6.2.1 Walkway Plates

The mouse requires a surface on which to walk that is transparent enough for the camera to capture the footfalls from beneath. Several different types of materials were tested for durability, transparency, and color filtering. The first walkway was built out of clear, 1/8" thick, Lexan plastic. The plastic surface ultimately ended up being too easily scratched and too thin; to the point of bending as the animal walked across. It was decided that a frosted material would provide optical filtering. Figure 18 shows a comparison of the clear plastic walkway and the frosted glass walkway. The frosted material allows objects in contact with the surface to maintain clarity and heavily filters out object not in contact.



Figure 18. Clear plastic walkway and frosted glass walkway.

It was decided that a green color filter would increase the contrast between the mouse's pink feet and white body. Figure 19 shows a comparison of live mouse images captured through various colored lenses. It was decided that the green and blue lenses increase the paw to body contrast, if only slightly. The blue filter required a larger intensity gain to be applied, as it was a darker lens, thus the green lens was chosen. Though the increase in contrast is not substantial, any small gain will prove valuable by reducing the necessity of the detection algorithm to digitally increase contrast later. The revised walkway was built from a frosted glass pane purchased from a local glass shop. To apply the green filter, a green lens was placed on the camera rather than trying to get colored and frosted glass. This proved to have the same effect and allowed for removing and swapping the lens as necessary during testing.



Figure 19. Frame Color Filtering Tests

6.2.2 Track Enclosure

The walkway and camera are mounted to a frame which also provides an enclosure for the track and mounts for the lighting and other external electronics. The track enclosure consists of two parts, the walkway section and the end caps. At end caps are small enclosures or "mouse houses" where the animal can be loaded or retrieved. Variations of these are shown in Figure 20.



Figure 20. Original (Top) and Refined (Bottom) Mouse Houses

The mouse house on the starting side is illuminated and the one on the ending side is shaded. This was done because the mouse should naturally run from light to dark. The houses themselves are spacious enough to accommodate the entire mouse. This aids in loading and removing the animal. The walkway section is only as wide as the mouse to help prevent the mouse from turning around and moving in the wrong direction across the capture area. The walkway is also enclosed with a clear plastic canopy to prevent the animal from escaping while still allowing the operator to see it. The canopy was tested with both a rectangular and triangular shape as shown in Figure 21.



Figure 21. Rectangular and Triangular Canopies

It was found that it is more difficult for the mouse to turn around when in the triangular canopy because it can't climb the wall like in the rectangular one. A cover, or hood, placed above the canopy and walkway, is also necessary to reduce ambient light and to provide a matching colored background for runs with light or dark mice.

6.2.3 Lighting

Many different types of light sources were tested in a variety of configurations. The light sources include ambient light (unlit), florescent cabinet lights, ultra-bright white LEDs, outdoor patio rope lights, high intensity fiber-optic point lights, halogen work lamps, and ultra-bright, wide-angle, LED light strips. These are shown in Figure 22.



Figure 22. Lighting Sources Tested

The florescent cabinet lights are 1.5ft long, 15 Watt, florescent lights purchased from a hardware store. They plug into a standard wall-outlet and can be mounted to any long rigid surface. The ultra-bright white LEDs are basic LEDs which emit a very bright white light. They consume about 2.3V and each requires a resistor to set the input current (~5ma). These can be mounted to any rigid surface and pointed in any direction. Many can be clustered together to increase the overall illumination. The fiber-optic point lights are from a table top device used to illuminate microscope slides or small work areas. This device has a high intensity light which is delivered to two bendable fiber-optic arms. This sits on the table top and each arm may be separately positioned to illuminate a particular area. The halogen work

lamps are standard 150 Watt lamps available at any local hardware store. These lamps plug into a standard wall-outlet and can be setup on the table top or mounted via the stand legs. They cannot be mounted any closer than 6" from the target because of the heat produced. The ultra-bright LED strips are professionally built strips of wide-angle, white, surface mount LEDs. The strips are available in 6", 1', and 2' lengths. Each strip operates at 12 Volts and consumes 60mA, 120mA, and 240mA respectively. They may be powered directly from a wall-plug adapter or from existing circuitry, provided it can supply the necessary current. The strips can also be chained together to share the same power source. The strips come with an adhesive backing for mounting to any flat, long, rigid surface and are thin enough to slide into brackets for mounting.

6.2.4 External Hardware

External interface hardware is necessary for power, control, and communications. The power circuitry consists of voltage dividers and current limiters to ensure the proper power level is supplied to each of the external components: microcontroller, trigger circuitry, and lighting. The triggering circuit consists of two LED and phototransistor pairs which create "trip beams" to determine when an animal has entered and exited the capture area so that only relevant data is captured. An output signal is activated when the first trip beam is crossed and deactivated when the second beam is crossed. This design uses MOSFETs to activate the sensor elements and an S-R latch to hold the output state. The circuit schematic is shown in Figure 23.



Figure 23. MOSFET Based Triggering Circuit

The components X1, X2, X3, and X4 are each n-channel enhanced MOSFETs. The part number used is Zetex ZVN3306A. The phototransistors, Q1 and Q2 are NPN phototransistor for visible and near-infrared detection. The part number used is Vishay BPW77NA. The triggering circuit was originally designed to be connected to the camera's general purpose input/output (GPIO) port. The GPIO pins are set to no-connect when disabled and are set to ground when enabled. It was discovered, though, that these pins were output only, so the circuit could be activated, but the trigger signal could not be read. The external hardware was, therefore, redesigned to simplify the circuit and to remove the dependence on the camera's limited I/O. The revised designed uses much simpler circuitry and includes a microcontroller to handle sensor I/O and PC communications. The revised circuitry consists only the two LED/phototransistor pairs and the necessary pull-up resistors and voltage dividers for each of the sensor lines to ensure that the 12 volts the trip beams use is converted down to the 3 volts the microcontroller requires. The circuit diagram for the revised design is shown in Figure 24.



Figure 24. Revised Microcontroller Based Triggering Circuit

The microcontroller used is an SiLabs C8051F300 processor mounted on a development board. The processor is based off an 8051 core and has 8 GPIO pins, UART communications bus, and 8kB of program space. The development board provides an RS-232 level shifter with serial port connector, a 12-pin header for connecting the sensor lines, and all the necessary power circuitry [25]. The code running on the microcontroller is a simple state machine with boundary checks. It reads sensor data from the two trip sensors and reports the state to the host computer. The triggering software has 4 states: idle, waiting, capturing, and done. On startup, the microcontroller is in *idle*. When a request to begin an experiment is received, the controller enters the *wait* state and activates the sensors and read the values from the first sensor. When the first sensor is tripped, the controller deactivates the LED to signal that it was crossed and then reads from the second sensor. This is the *capturing* state. When the second sensor is tripped, the controller deactivates it to signal that it was crossed and transitions to the *done* state. This tells the host computer to stop the image capture from the camera. The microcontroller is programed to only report the current state when queried, so it dependent on the program running on the PC to initiate all communications. Using the microcontroller in this way helps to keep the external circuitry from becoming too complex and eases the burden on the PC when capturing data.

6.2.5 PC Capture Software

The PC capture software incorporates a number of programming elements which aid in the high speed capture and storage of image data, the communications with the camera, and the playback of the data stored. The capture software itself is written in C++ using Microsoft's Visual C++ Studio .NET Addition. Visual Studio provides all the basic code which handles the creation of the program window and all Windows event handling. The interface to the camera is controlled by the PixeLINK API Library. This is an external library available as part of the PixeLINK SDK and is a collection of API functions. The API functions are medium level routines which provide all the basic commands to the camera, ranging from setup parameters to data acquisition. As the library is simply a collection of API functions, a wrapper class had to be developed to properly incorporate the functions into the C++ object oriented environment.

During the capture process, all of the available system resources are devoted to acquiring and storing the image data as rapidly as its coming in so as to avoid missed frames. Typically, a Windows program only executes in a single thread. If a single thread was used for this application, then during the capture process, all of the controls would become frozen. This would mean that until the second trip beam was crossed, the program would continue to record data, without giving the user the ability to cancel or reset. To circumvent this, the capture software is written to be multi-threaded. In total, the software has three separate runtime threads. The main thread draws performs all the vital program functions and provides for user control. When told to begin a capture sequence, the program starts a secondary thread which opens the serial port and handles all communications with the external hardware. This thread posts messages back to the main thread alerting it to what stage of the capture process the system is in. When the first beam is tripped and the main thread is alerted, another secondary thread is started which actually reads the data from the camera and saves it, frame by frame, to the hard drive as a video file. This thread is set to a higher priority than the other two as it needs to keep up with the rate at which the camera is delivering data. At this point, all three threads are running; the main thread to allow for user intervention, the communications thread to watch for a change in the external hardware state,

and the capture thread to acquire and store data from the camera. When the second trip beam is triggered, the communications thread will post a message to the main thread to end data capture. The main thread then stops both the capture and serial threads and the system is idle again.

The management of the image data is performed by a third-party ActiveX class called VideoOCX which was written by Marvelsoft to provide quick and easy integration of video into C++ programs. VideoOCX manages the storage, compression, and playback of the video. The structure of data coming from the camera matches the structure received by the ActiveX function so the incoming data is simply passed directly to the ActiveX control where it is then put in to the standard audio/video interweave (AVI) format and saved to disk. Once capture is complete, the video data can be reviewed in a window provided by the ActiveX class.

6.3 **Prototypes**

The following sections document the prototype designs which were developed. Each stage shows marked changes in design and technology. The first prototype is basic system used to test the feasibility of the camera/plate design. The final prototype is the result of extensive research and many hardware iterations and user reviews. It represents a fully functional solution capable of meeting all the design requirements and collecting real experiment data.

6.3.1 "Pagoda" Proof-of-Concept Testbed

The first test setup was built to accommodate a camera capturing video under a transparent surface that the animal moves across. The physical setup is comprised of a plastic and wooden frame with a black covering to prevent interference from ambient light. The camera is mounted inside the frame, facing up, where above it, a piece of transparent plastic is placed. A hood was also built which straddles the sides of the box and is used to block overhead lighting from being seen in the camera's view. The complete assembly of the Pagoda stand is shown in Figure 25.



Figure 25. Pagoda Camera/Plate Test Stand

The first tests were conducted prior to building the Pagoda stand to determine the working distance between the plate and the camera. To do this, the Quickcam camera was mounted to a shaft were the vertical distance could be varied. A ruler was placed underneath the camera and the distance between the two was adjusted for different pixel per centimeter ratios. The results of this are shown in Table 2. It was concluded that 50pix/cm with a window size of 12.8cm x 9.6cm at a working distance of 17cm was a good balance of settings for testing and the Pagoda stand was built to these specifications.

Working	Resolution	Active Area	Active Area
Distance (cm)	(pix/cm)	Length (cm)	Width (cm)
23	35	18.2	13.7
21	40	16.0	12.0
19	45	14.2	10.6
17	50	12.8	9.6
15	55	11.6	8.7
13	65	9.8	7.3
11	80	8.0	6.0
10	90	7.1	5.3

Table 2. Quickcam Working Distance and Resolution Comparison.

The Pagoda stand was used to conduct the initial lighting tests. The first tests used only the florescent cabinet lights and the outdoor patio rope lights. These early tests also quickly showed the need for a hood over the capture area to block out overhead lighting. The florescent lights were tested in a variety of locations. They were placed inside the frame, on the bottom, flanking each side of the camera, inside the frame, on the top, just under the walkway plate, and outside of the frame, on the top, just to each side of the walkway plate. These tests were done to try to obtain the best amount of contrast between the mouse's paws and its body and the background.

The Pagoda stand was also used to test if a standard camera's frame rate of 15fps at full resolution would be sufficient for data capture. At 15fps, the mouse can move a paw from one position to another with the video capturing little or no transition. Figure 26 demonstrates this by showing three successive frames recorded at 15fps. It is shown that contact and swing durations are each represented by a single frame. In this video, the mouse is moving at 21cm/s. If the mouse were moving at a higher velocity, a contact or swing could be lost altogether. At 30fps, for the average mouse speed, this only amounts to approximately 10 frames per complete stride or 2.5 frames per contact or swing.



Figure 26. Successive Frames Captured at 15 fps

6.3.2 "Poplar" Device Parameters Testbed

The second prototype system, shown in Figure 27, was built to utilize the PixeLINK PL-A741-R camera and to allow for hardware reconfiguration to test a variety of design parameters. This stand was built out of ¹/₂" thick poplar wood. This provided a large and sturdy frame for any number of hardware components to be easily mounted. The primary considerations for this phase of development were camera placement, walkway materials, lighting schemes, and enclosure configurations. Extensive testing was done, particularly on the lighting schemes, to obtain images with the highest contrast possible.



Figure 27. Poplar Gait Stand

Camera placement, or more specifically, the working distance between the camera lens and the walkway plate, was the first design aspect which needed to be addressed as it determined the physical dimensions of the stand. A resolution of 100 pixels per inch was chosen as it had previously been determined to provided adequate definition of paw rotation and the spread of digits. Given this resolution, and the selected 8mm lens, the working distance was set to 27.4cm.

In this design, the walkway plate had to be reconfigurable so that a variety of materials with various properties could be tested quickly and easily. To accommodate this, the stand was designed so that the walkway sits on two long support rails, one each on the left and right side, at the precise working distance from the camera lens. The rails are covered with a strip of foam insulation to both grip and protect the walkway.

The need for reconfigurable lighting schemes was the topmost requirement for this prototype phase. Illumination of the capture area plays an essential role in the quality of the data gathered. To support this requirement, lighting mounts for the cabinet lights, and later for the LED light strips, were placed directly under the walkway, parallel to the plane of the walkway, and directly above the walkway. Lights were also able to be positioned anywhere on the base of the stand. The various lighting mounts are shown in Figure 28. The image on the left shows the parallel and above surface mount areas. The center image shows the a florescent light mounted under the walkway. The right image shows the LED strip mount under the walkway in place of the florescent light.



Figure 28. Poplar Stand Lighting Mounts

Finally, this prototype had to facilitate the use of actual mice to gain real experiment data. To accommodate live animals, an enclosure was built around the walkway which incorporated two mouse houses, light and dark, a triangular, transparent canopy, and a dual
sided hood. The hood is a piece of plastic which spans a little over the full length and width of the capture area. It is coated in white on one side and black on the other to provide the appropriate matching background for when light or dark mice are run. Testing of the Poplar prototype consisted mainly of developing an optimal lighting scheme. Some video capture triggering tests were performed before shifting to the next prototype phase.

Lighting Tests

Proper illumination of the capture area is paramount to getting usable data. The white mice have a white body with light pink paws, nose, and tail. When exposed to overhead illumination, the background appears very bright and the body and appendages appear very dark. The pink areas do not differ in intensity enough from the body to discern one from the other. Conversely, when illuminated from underneath, the body and appendages appear very bright and the background appears dull. This makes the white mouse body standout, but the paws turn out to be the in the same intensity range as the background. Because of the low contrast between the paws and either the body or the background, simple image thresholding is not a viable identification technique. To assist later on with the paws and the rest of scene.

In the very first tests, the walkway plate was a transparent pane of $\frac{1}{8}$ " inch thick plastic and there was no filtering of the light. The plastic, however, was easy to scratch, too flexible, and too thin to transmit light parallel through its structure. It was decided to switch to a $\frac{1}{4}$ " glass pane instead. The increased thickness would allow for the plate to be illuminated internally, which was one of the new lighting schemes under development. It was also decided that the glass be frosted as a means of optical blurring so that only objects touching the surface were in detail. Figure 29 shows the result of the optical blurring. The front-right paw in contact with the plate appears clearly. The front-left paw not in contact with the plate appears only as a dark shape.



Figure 29. Results of Various Lighting Schemes



Figure 30. Optical Blurring Effect Due to Frosted Glass

Multicolor lenses were also tested at this time to see if color filtering would help enhance the contrast. Red, orange, yellow, green, blue, and polarized lenses were each tested. While there were not significant differences, the green lens did provide some contrast enhancement, so it was decided to continue using it with every subsequent experiment.

The lighting tests on the Poplar stand were extensive. Every light source was tested in multiple configurations. Figure 30 shows varied results of these tests. Image A is the result of a low intensity light shining up from under the walkway. The overall intensity is dull causing the paws and the background to share the same intensity range. Image B is the result of bright lights being placed above and to the side of walkway and a digital gain added to increase intensity values. Although the paws are particularly clear in this image, the outline of the body is also very distinct. When performing object detection, the body and nose are very difficult to separate from the paws. Image C represents bright lights placed below the walkway illuminating both the mouse and a white backdrop that has been placed over the canopy. A digital intensity gain has also been applied. Although the body intensity is closer to that of the background than in image A, the contrast between the two is still greater than the contrast between the body/background and the paws. Image D is the results of lights being placed exclusively above the walkway and a black backdrop placed on the canopy. Only the sides of the animal are illuminated which creates a strong body and tail outline, even through the frosted glasses. It still remains difficult to distinguish the paws though.

The cabinet lights were tested in the same three positions as in the Pagoda setup: next to the camera under the walkway, to each side directly under the walkway, and to each side directly above the walkway. A forth position was also tested with the Poplar setup; parallel with the edge of the walkway such that the light is directed directly through the plate. This was based off the Clarke and Still setup [8], [9]. The idea is that any point of contact the animal makes with the plate is illuminated while the rest of the scene remains dark. The cabinet lights were tested first in this configuration. It was quickly shown though that they did not have enough power to provide light throughout the entirety of the plate. The edges were well lit, but the falloff toward the center of the plate was substantial. There was also not sufficient illumination of the points of contact to differentiate anything from the dark background.

Next, the cabinet lights were setup to illuminate both the edge of the walkway and the area above it. This resulted in the outline of the animal becoming very apparent, but there was still too little contrast between the paws and the body. After that, the cabinet lights were moved down to the base of the stand and directed upwards. This had the effect of lighting the paws and the body, but leaving the background dark. The paws and the body were again very similar in intensity, this time being similarly bright rather than dark.

At this point, lighting tests switched to using ultra-bright white LEDs. Twenty LEDs were placed, ten on each side, along the edges of the walkway. Like the cabinet lights, these did not provide enough light to illuminate to the center of the walkway. They also cast a very non-uniform falloff pattern in the image. The LEDs were also tried along above the walkway, but the results were the same as the cabinet lights: bright background, middle intensity body and feet.

At this point, mounts for the cabinet lights were made directly underneath each side of the walkway. Tests from this configuration showed the greatest amount of intensity contrast. The paws were in high contrast against the white body and had enough contrast against the mid-range background that they were able to be selected out using the image processing

techniques concurrently being developed. This setup was used until it was decided that there was simply too much non-uniformity of the intensities in the image data. The image matching algorithms could detect the feet when they were properly illuminated in the center of the capture area, however, there was too much variation in the intensity on the boarders for reliable detection. At this point, new lighting schemes were once again explored.

It was decided that if the capture area could be flooded with light that it would saturate the camera except for where the paws were located and the detection algorithm could pick and choose from whatever detail remained. The first test of this was with the fiberoptic point lights. One of each of these were positioned at each end of the track and pointed up toward the walkway from underneath. It was found, however, that they did not provide a wide enough beam to light the entire area without falloff. Next, two halogen work lamps were tested. The provided more light than any of the other sources, but were too bulky to place under the walkway or mount in any reasonable way. They also produced an excessive amount of heat which was undesirable around the camera and control electronics.

Finally, the ultra-bright white LED lighting strips were tested. At the time, they were only a recent acquisition but proved to be an excellent solution to the lighting issue. The light strips were precisely spaced, ultra-bright, 90° spread, white LEDs mounted to a thin rigid strip measuring 12" long. Two of these strips were mounted directly underneath the walkway, in place of the cabinet lights. The light provided from the strips was bright and uniform, with the exception of some slight falloff at each end.

Triggering Tests

For the early tests of the camera, the video was captured for a set period of time. This worked fine for testing, but files were large with long spans of empty frames, and often the run was cut short because the time ran out. For acquiring usable data, a start/stop triggering system had to be developed. The first design of the triggering system utilized the camera's existing general purpose input/output (GPIO) ports. A circuit was designed which would signal when each of the trip beams were crossed and, therefore, when the animal was in the

capture area. The circuit was built to allow the camera to activate/deactivate the entire circuit so the trip beams would only turn on when an experiment was being conducted. After the circuit was built and connected to the camera, it was then found that the camera's GPIO ports were, in contrast to the documentation, output only ports. To attempt to bypass this issue, the signal line was connected to the external hardware trigger pin on the camera and the camera was set to external trigger mode from internal trigger mode. Normally, the camera generates its own internal periodic signal to capture a frame of data a the appropriate time. This can be overridden with an external device which connects to the external trigger pins. To interface the existing trip beam circuit, camera was set to watch the external trigger pin and when it saw the first switch in value (i.e. the mouse crossed the first trip beam), instead of capturing a frame, the camera switched over to the internal periodic signal and then proceeded to capture each frame of data. To stop the capture, instead of using the second trip beam, the software watched for a change the last column in the image data which would indicate that the mouse was present and that recording could stop. In practice, this method was very processor intensive as it require multiple program threads to constantly be watching the image data without resource conflicts. It was decided that a redesign of the external hardware was needed to simplify the system and remove the dependence on the camera's limited I/O ability.

The revised designed uses much simpler circuitry and includes a microcontroller to handle sensor I/O and PC communications. The revised system still uses the two LED/phototransistor pairs, now connected the microcontroller. The microcontroller keeps track of which beams are tripped and if they are tripped in the correct order. This information is then passed along to the PC via a serial connection. This setup has the advantage of being easily reconfigured, as all that is needed is to reprogram the microcontroller, and does not require the PC software to keep track of the external hardware state. The is the responsibility of the PC software though to query the triggering system and to do so in a timely manner. It also means that the PC must be equipped with a serial port.

On-Site Tests

Once the lighting and triggering issues were worked out and the software written, the system was ready for the first live animal tests. The initial tests successfully captured data and demonstrated the successful implementation of each of the subsystems. The success of this prototype warranted the design of a revised system which would incorporate modifications based on operator observations, improvements to the lighting scheme, and an overall streamlining of the system.

6.3.3 "Fleet Foot" Complete System Implementation

The final prototype system, shown in Figure 31, is an updated, streamlined version of the previous Poplar design. The goal of this design phase was to build a system which retained the basic features of the previous design, but to implement them in an improved or finalized form. The stand is now built from Delrin plastic rather than wood. Delrin is as sturdy and can be machined as easily as wood, but is non-porous, which allows for easy cleaning. There is still a glass walkway mounted above the camera with two enclosures on each end for the animal to run between. The same microcontroller based triggering system is used and the LED strip lights are now permanently mounted directly under, and to each side, of the walkway. Further improvements and revisions will be detailed in the following sections.



Figure 31. Fleet Foot Gait Acquisition System

Camera to Plate Working Distance Modifications

Feedback gained from the previous design indicated that the overall height of the device was too large for tabletop use. It was therefore decided to reduce the total height by approximately10cm ($^{1}/_{3}$ the height). This gives a working distance of 17.5cm and has the additional advantage of increasing the resolution of the image to roughly 150 pixels per inch. However, with the increase in resolution, there is a decrease in frame rate. The system is now only able to reach a maximum of 75 fps. The other drawback of this modification is that there is now more "fish-eyeing" of the image and the intensity values. This means that near the ends of the image there is increased image distortion and variation in light intensity. This effect is shown in Figure 32 and is caused by using the lens so close to its minimum working distance.

Figure 32. Intensity "Fish-Eye" Effect

Walkway Modifications

The walkway on the previous prototype was 12" long to match the length of the desired capture area with the trip beams positioned directly outside the houses on each end. Testing, though, showed that the mice would often walk part ways out of the house and then stop to determine if it was safe to cross to the other end. This would trigger the system to start recording before the mouse was actually in motion. To correct for this, the walkway, still a frosted pane of 1/4" thick glass, has been extended to 20" in length. There were two main reasons for this. First off, the extended length gives the animal time to come out of the house and explore a bit without breaching the trip lines and starting data capture. This way, when capturing is triggered, the animal will already have begun walking in a standard gait toward the other end. This increases the usefulness of the data and reduces blank or irrelevant frame from the video. The second reason for extending the walkway plate is that it now forms the floor of the light side house as shown in Figure 33.



Figure 33. Light Mouse House with Translucent Floor

It was decided that in the previous version, even though the light side house was not lined in black, like the dark side one, the light side house was still not bright enough compared to the well lit walkway to promote the animal run out and across. With the glass plate now extended under the light side house, this allows the lighting to be extended underneath it as well, making it as bright as the rest of the walkway. It also makes the surface uniform, so the mouse is less aware of the transition from the house to the walkway. Testing shows the advantage of this modification as the mice are now much less reluctant to run to the other end of the device. The width of the plate was changed as well. The plate on the Poplar prototype was 6" across, but the canopy only sealed in a 2" wide space. Since that much width was unneeded, the walkway was reduced to 3.5" across. This gives a 3" wide walkway to better accommodate injured animals and a 1/4" lip on each side for mounting.

Lighting Modifications

The lighting scheme for this prototype is based off the best of the various types tested in the previous design. The ultra-bright white LED lighting strips are used and are mounted directly below the walkway plate, just to each side of the edge of the capture area. These provide ample uniform light across the entire region. The change from the last version to this version is that they now extend the full 24" length of the device. This is to ensure that there is uniform light well past the leading and trailing edges of the capture area, as well as serving to illuminated the bright side mouse house.

Enclosure Modifications

The enclosure setup for this prototype is a finalized version of the experimental design from the Poplar prototype. Changes were made to the design of the houses, the way the light and dark side houses were differentiated, the shape of the canopy, and the way the walkway is mounted to the stand. User feedback suggested that to ease the loading and removal of the animals that the houses should be top loading rather than side loading. The houses were therefore built with a removable cover on the top rather than a hinged door on each end. In the previous design, the house were each built identical and then the interior of the dark side house was painted black. For this design, since Delrin plastic is available in white and black, the light side house was built out of white plastic and the dark side house out of black plastic. This allows the light side house to be very bright and the dark side house to be very dark, much more than either were in the Poplar prototype. With all the modifications to promote better motivation of the animal to move from one end to the other, the canopy style was changed from a triangular shape to a rectangular one. The reasoning for this is that it allows more freedom of movement for injured animals. It also provides a stable platform for the hood to rest on. The final modification is to the way the walkway plate is mounted to the stand. Rather than resting on the supports, the walkway now slides into channels on each side of the stand, as shown in Figure 34. This provides better support for the glass plate while still allowing it to be removed for cleaning or swapping. The black plastic floor of the dark side house also slides into this channel.



Figure 34. Walkway Insertion Channels

Modular Design

One of the new aspects to the Fleet Foot prototype is its modular top and bottom design, shown in Figure 35. In this design, the top part of the stand, which includes the walkway, lighting, and enclosures, separates from the bottom where the camera and all the electronics are mounted. The advantage of this is that the top can be removed for animal training, while the more expensive and less portable components can remain in one safe location. This also allows for multiple modules to be used in varying locations or simultaneously. Additionally, this can produce a better response from the mice as the same setup they were trained can also be used when real experiment data is collected. When used separately, the top plugs into its own power adapter. When joined with the base, connectors on the support struts link up the sensors and lights.



Figure 35. Modular Halves of Fleet Foot Stand

Preliminary Tests and Results

The first tests with the new prototype resulted in successfully acquiring usable data and spotlighted a few more design issues to be addressed. User feedback indicated subtle changes to the physical design while data analysis showed the need for further modifications to the lighting scheme. The first change requested to the physical setup was to make the mouse houses removable to help with cleaning. At the time of testing, the houses were fastened directly to the stand which made it difficult to reach in and clean up anything the mice left behind. The next recommendation was to switch back to the triangular shaped canopy. The rectangular canopy allowed for too much freedom to explore the runway, so the animals were prone to wander around rather than run straight through. It was also requested that the hood which provides the light or dark background and blocks overhead light be affixed to the stand by a more stable means. At the time, it was only resting on top of the canopy and could easily be bumped off. Also, with the move back to the triangular canopy, the hood had to be supported by some other method. Lastly, though this was not a user issue, it was decided that it would be preferable to hinge the covers of the houses simply so there were two less independent parts to keep track of.

Lighting again became an issue with this implementation. The decrease in the working distance between the lens and the plate caused an increase in the fish-eyeing of the light intensities (See Figure 32). This is due to the curvature of the lens, the close proximity to the plate and light source, and wide angle viewing. This creates a problem with setting a correct overall gain, or median intensity. If the gain is increased, the ends will brighten up, but the middle becomes washed out, resulting in less definition of the paws. This is shown in image A of Figure 36. If the gain is decreased such that the paws are no longer washed out in the center region, then the intensities of the paws and the background, at the two ends, become much closer, again resulting is less definition of the paws. This is shown in image B of Figure 36.



Figure 36. Paws Washed Out and in Dark Edge Region

To combat the issue of non-uniform background intensities, for first tests of this system, the paws of all the animals were inked with a dark indigo ink and the gain was increased to make the ends nearly as bright as the center. This had the advantage of having the white

body blend into the white background and provided stark contrast between the paws and the rest of the scene. Figure 37A shows the first trials with inking. The gait was not increased to match the body to the background because it was found that, in practice, the edges of the paws would washout. This resulted in smaller looking paws which were unable to be detected. This was just one of the disadvantages to this method. There were several more found. The first being that each animal had to be inked before the experiment. This proved to be time consuming and messy. Also, it is quite difficult to ink a mouse's paw, particularly without getting ink on any part of the body. What typically resulted was the mouse's belly also being dyed black. Often times too, the ink was not entirely dry when the mouse walked across the device and footprints were left behind, sometimes dark enough to register as contact point even after the mouse had moved on. Despite these issues, inking the paws of healthy mice proved to be the best method to create a high amount of contrast and acquire usable data. Once the mice were injured, however, the limitations were magnified. The mice would drag their entire body through the ink and there was little way to distinguish a paw from the rest of the amorphous dark area in the image. An example of this is shown in Figure 37B. To solve this problem, and to remove the dependence on inking the paws altogether, it was decided to mount a second set of LED light strips just above the walkway to help brighten the background of the image without having to adjust the gain applied to the intensity data. The physical design modifications and the changes to the lighting scheme resulted in a second version of the Fleet Foot prototype being developed. As the changes were only minimal, there was no need to create an entirely new prototype.



Figure 37. Mouse with Inked Paws Before and After Injury

6.3.4 "Fleet Foot" Complete System Implementation - Mark II

The Mark II version of the Fleet Foot system, shown in Figure 38, represents the most current design. All of the recommended modifications have been applied resulting in a streamlined device. This system represents the culmination of research conducted on the mechanical ascpects of the system.



Figure 38. The Fleet Foot Gait Acquisition System - Mark II

Mark II Enclosure Modifications

The first revision to be implemented was the ability to remove the houses on each end for cleaning. In the previous version, the houses provided the cross connection between two sides of the upper half of the stand. For the houses to be removable, connecting braces were added between each of the struts to hold the two side together, as shown in Figure 39A. To hold the houses in place, but still allow them to be easily removed, it was decided to embed magnets into each of the ends and the base of the houses. This way the houses can be clicked into place when in use and easily pulled off for cleaning. This is shown in Figure 39B. The back wall of the house extends down to touch the rails on each side to provide added

stability when the house is attached the stand. The covers of the houses are now attached and hinged as well, to allow easy access without the issue of being misplaced. Additionally, the canopy has been reverted back to the original triangular shape. This proves very effective at keeping the animals on track and not turning around mid-run or wandering about. The hood was also modified to have extenders with pegs on each end which are inserted into slots in the top of each of the houses. The hood still retains its two-sided design for use with either light or dark mice. The pegs are spaced in an offset configuration to ensure proper connection when the hood is flipped. The new hood design in shown in Figure 39, images C and D.



Figure 39. Fleet Foot Mark II Enclosure Modifications

Dual Level Lighting

The most important modification to the Fleet Foot system is the dual level lighting scheme. To ensure a white background when using white mice for the best contrast between the paws and the rest of the scene, two extra ultra-bright LED strips are mounted above the walkway, one on each side. The light strips are mounted at a 45° angle pointing up toward the hood. This way, the background is brighten without adding too much extra brightness to the body and paws. The intensity of the lights is trimmed using a 570 Ω resistor in series with the strips. This is done to balance the background intensity with the rest of the scene. The dual level lighting system is shown in Figure 40 and a comparison of the image data with and without the dual level lighting is shown in Figure 41. In Figure 41, the bottom image is a frame captured without the dual level lighting. The top image is a frame of the same mouse captured with the dual level lighting. With the dual level lighting, the body blends into the background and the paws remain in comparatively high contrast.



Figure 40. Dual Level Lighting



Figure 41. Dual Level and Single Level Lighting Comparison

System Tests

The Fleet Foot prototypes were tested by acquiring real experimental data for a set of 13 mice over a four week period. The first week runs were conducted with healthy mice on the first version of the Fleet Foot system. The subsequent three weeks of experiments were conducted using the Mark II version of the system. The system performed very well during all tests and is suitable for continued research. The experimental results will be presented in Chapter 9.

6.4 Final Design Discussion

Overall the camera-plate solution works well. The main difficulties lie in determining when a paw is in contact with the plate. The low contrast between the paw and the rest of the image is at the core of this problem. It was through a combination of optimal lighting schemes and digital enhancement that points of contact were actually able to be determined. The resolution of the data is much greater than any other solution, but at the expense of being able to collect force data. With all the subsystems integrated and tweaked, the data collected is of usable quality. The final prototype has been shown to stand up to repeat use and to collect consistent data. Conducting an experiment simply involves loading the animal into the bright side house, setting the hood to the proper side, depending on the color of the mouse, and starting the capture process from the PC software.

6.4.1 Physical Dimensions

The final prototype of the gait acquisition system is a table top device measuring 24" long, 6" wide, and 14" high. When separated into its two modules, the top walkway modules has a height of 8.5" and the bottom component module has a height of 5.5" with both retaining the same length and width.

6.4.2 Interface and Power Requirements

The final prototype of the gait acquisition system requires a PC running Windows 2000 or greater which meets the following minimum requirements: processor speed of 1.6GHz, 512MB RAM, 5GB of free hard drive space (for storing raw experiment video data), a screen resolution of 1024x768. The PC also must have a dedicated Firewire port (cannot be a combination USB Firewire card) and a free serial port. The final prototype requires a 1 amp, 12 volt, regulated power supply with a positive center adapter.

7 IMAGE PROCESSING TECHNIQUES

7.1 Introduction

Gait analysis is conducted on a PC via a custom designed detection algorithm. The algorithm utilizes a myriad of image processing methods to filter, analyze, and report relevant paw contact information from the visual capture of the gait. This chapter will detail the requirements the algorithm must meet, several image processing techniques to filter, categorize, and match image data, and the implementations designed and tested.

7.2 Locating Relevant Data

The first step to paw identification is examining the data and extracting the information relevant to object matching. There are several methods of determining which points can be useful in the image. During the course of this research, three of these methods were explored; threshold separation, edge detection, and difference of gaussian.

7.2.1 Separation by Threshold

Threshold separation is an image processing technique where relevant data is extracted from background data by partitioning the image data based on an intensity threshold value. That is, any point in the image above a threshold is copied into the new data image (7.1).

$$D(x, y) = I(x, y) > Thresh$$
(7.1)

Examples of threshold separation at various threshold values are shown in Figure 42. Basic threshold separation when the data is separated into two groups based on a single value or a slightly more advanced method where there is an upper threshold and a lower threshold. This requires the object in the frame to be in high contrast with the background, otherwise, it is difficult to pick a threshold value where the separation always holds true.



Figure 42. Basic Thresholding at Incremental Levels

Foreground extraction is an advanced method of threshold separation where a threshold mask is applied to the image to emphasize relevant data. This is done by computing a background image either by performing a heavy blur to the original image or by capturing an empty frame. The background image is then subtracted from the original image. The result is an image containing only the foreground objects. This method is also useful for removing undesirable artifacts in the data. The disadvantages are that it amplifies any noise present in the image and that the resulting image is of lower contrast than the original. This method can be extended for use as a filter to prepare an image with a nonuniform background for further image processing, as shown in Figure 43. First, the mean intensity(μ) of the background image is computed. An inverted intensity mask is created by subtracting the each pixel value of the background from the mean (7.2a). The mask is then added to the image data to produce a result which has a uniform background (7.2b).

$$U(x, y) = \mu - B(x, y)$$

$$D(x, y) = uint 8 [I(x, y) + U(x, y)]$$
(7.2)



Figure 43. Foreground Extraction Using Background Normalization

7.2.2 Edge Detection

Edge Detection is an image processing technique where the edges of objects in the image are detected and outlined. Edge detection works by finding the areas in the image data where there are sharp transitions in intensity. Basic edge detection is performed by simply taking the derivative of the image and highlighting all the points which are above a set threshold. More advanced methods use multiple thresholds or higher order derivatives. Three different methods of edge detection were explored during the course of this research. These are the Sobel / Prewitt method, the Canny method, and the Zero-Cross method.

Sobel / Prewitt Method

The Sobel / Prewitt method of edge detection computes the image gradient, or two dimensional derivative of the image data using either the Sobel (7.3) or Prewitt (7.4) operators.

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$
(7.3)
$$\begin{bmatrix} -1 & -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$
(7.4)

The rows and the columns are computed separately by performing a 2-D convolution between the corresponding operator matrix and the image. The two results are then added together to form the final image containing the edge data. The only difference between the two operators is the Sobel method will slightly accentuate vertical and horizontal edges.



Figure 44. Sobel and Prewitt Edge Detection

Figure 44 shows the results of the Sobel and Prewitt methods on a low contrast, non-uniform intensity image. The first frame is the original image. The second frame shows the result of the two detectors. The values in the image are equal to the 2-D gradient magnitude. The third frame shows a inverse of the gradient magnitude to better highlight the edges as well as the noise. The final frame shows the results after performing a black and white threshold of the gradient magnitude. All intensities higher than 75 (scale: 0-255) were kept. This gives the final edge detection result. It can be seen that the Sobel method produces slightly more intense edge segments.

Canny Method

The Canny method of edge detection is used when there are discontinuities in the image intensities. This method uses two thresholds to distinguish between strong and weak edges. The weak edges are only included if they connect with a strong edge. The process begins by first blurring the image with a gaussian filter to remove high frequency content (7.6). This is done to remove noise from the image which may be identified as a weak edge. The image gradient is then computed using the any of the basic edge operators, though typically, the Sobel operator is used. The raw edges from the resulting data are then thinned to a single pixel wide line segment. To thin the edges, each pixel is compared to its neighboring pixels along the direction of the local gradient. If the pixel value is greater than that of its neighbors, then it is copied into a new image containing all the local maxima. Two threshold levels are then applied to the new data set. The larger of the two thresholds identifies all of the strong edges. All pixel values above this threshold are copied over into a strong edge image. The second, smaller threshold identifies weaker edge segments. All pixel values above this threshold are copied into a weak edge image. The final step is to link the strong edges with the weak edges to provide one continuous edge. This is done by selecting a pixel in the strong edge image and tracing along the edge by searching for adjacent pixels. When no more adjacent pixels are found, the same neighborhood in the weak edge image is searched. If an edge pixel is found, it is added to the strong edge image and the search continues. The result is that anywhere there is a break in the strong edge, and there's a weak edge present, the weak edge data fills in the gap, and all weak edges not touching a strong

edge are filtered out. Figure 45 shows the results of Canny edge detection on a single frame of data for incremental thresholds. It is shown that lower thresholds locate more edges but are much more susceptible to detecting edges in the background and noise. The high thresholds remove the weaker background edges, but also remove details which are still desired.

There are several advantage of this method over the single threshold methods. The gaussian prefiltering makes it less susceptible to identifying false edges due to noise, the thinned edges provide a more accurate boundary for subsequent image processing procedures, and the dual thresholds allow for low contrast edges, which may otherwise not be located, to be added to the data set without adding extraneous data. The drawback of this method is that it is computationally intensive and requires a lot of memory space to store intermediate data in.



Figure 45. Canny (Left) and Zero-Cross (Right) Edge Detection Results at Varying Thresholds

Zero-Cross Method

The Zero-Cross method is another means by which to perform edge detection on images with non-uniform intensities. This method computes a second-order spatial derivative to identify areas where there is rapid change in image intensity. Any point where the gradient quickly spikes and then quickly changes back is considered an edge. This can be represented as zero-crossings in the second derivative. To find the edges, the high frequency noise is first removed using a gaussian filter, as is done in the Canny method. The second order derivative is approximated by convolving the image with a Laplacian operator. The most common Laplacian operators used are shown below.

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$
(7.5)

The resulting image has values of zero where the change in intensity is gradual enough not to be picked up; negative values on the brighter side of an edge; positive values on the darker side of an edge, and a ridge of zeros between the positive and negative values where the edge lies. The edges are extracted by searching the data for all positive to zero to negative transitions above a given threshold of change. This method has the advantage of being the most robust to non-uniform intensity shifts in the data. The intensity change across an edge does not have to be consistent throughout the image. Only the transition must be uniform, independent of intensity value. The drawback of this is that for low contrast edge detection, where this method would be typically used, it is much more susceptible to noise. Figure 45, on the right side, shows the results of Zero-Cross edge detection on a single frame of data for incremental thresholds. It is shown that a low threshold produces more edge points, but many of these stem from noise in the background. Higher thresholds quickly remove much of the noise edge points but also result in creating discontinuous object edge lines. This becomes a problem if a closed loop edge is needed. Pre-filtering the image and selecting an appropriate threshold is, therefore, crucial. This method is less computationally intensive

than the Canny method as it does not require as much scrutinizing of the data to determine an edge. The Sobel / Prewitt methods are still faster since they do not require a second pass of the data.

7.2.3 Difference of Gaussian

The Difference of Gaussian (DoG) is a processing technique which accentuates textures within an image. This is performed by passing the image through a Gaussian filter several times and then computing the difference between each iteration [26], [27].

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad \text{for } x, y = -n, ..., -1, 0, 1, ..., n$$
(7.6)

where there are (2n+1) rows and columns

The result is that uniform regions or regions with slowly changing intensity are filtered out and areas with strong inconsistencies in intensity, or high in texture, are strengthened. The first step is to set up a common Gaussian filter. The sigma value of the Gaussian function controls the amount of smoothing between each iteration. A smaller sigma will allow finer textures to be accentuated, but is also more susceptible identifying noise as texture. To produce the DoG data the image is convolved with the Gaussian filter and stored. The result is then convoluted again with the Gaussian filter to effectively produce an image with a 2σ blur. The first blurred image is then subtracted from the second blurred image to produce the DoG. To produce multiple levels of DoG's simultaneously, this process can be repeated until the desired number of levels are computed.



Figure 46. Four Levels of Difference of Gaussian Filtered Image

Figure 46 shows the first four levels of a Difference of Gaussian operation. The first level detects the background noise as texture and so, while the paws are detectable through the noise, its not yet the best solution. Level 2 removes much of the noise and the remaining data points correspond to the paws in contact with the plate, the outline of the animal, and smudges on the glass. The third and fourth levels help to further filter non-contact points and extraneous objects.

7.3 Categorizing Data

Categorization is a necessary step toward assigning value to relevant data extracted from an image. This can be done by grouping similar pixels of data together to form a higher level data set, by describing a region of data by its relation to other similar pixels or to an external reference, or by finding robust key points exclusive to the data set.

7.3.1 Shape Clustering

Shape Clustering is a method used to link closely related pixels together to form a larger object. There are two primary method of grouping data into shapes. Groups can be formed by proximity or by connectivity. Proximity grouping states that if one pixel is within a certain range from another pixel, the two can be associated. This can be determined by calculating the distance between the two pixels and checking the result against a set inclusion threshold, or by searching within a boundary around the target pixel and

associating any other pixels found. Additionally, proximity clustering can be used to group together clusters of pixels which have already been associated with one another to form a single, larger object. This method has the advantage of being able to link together discontinuous portions of a single larger object. The disadvantages are that outlying pixels will be missed and that two separate objects located within the proximity threshold distance of each other will be joined together.



Figure 47. Grouping Pixel Regions by Proximity

Figure 47 shows proximity grouping of segmented pixel regions which all belong to a single paw. A search radius of 15 pixels is used to join together the regions and to exclude the extraneous object found in the bottom left.

The second method of shape clustering is by connectivity. Connectivity grouping is when one pixel is found and forms a new group, then adjacent pixels become associated with the first one, and then pixels adjacent to those become associated with the set [28]. This continues recursively until all connected pixels are grouped together. Connectivity grouping can get very computationally expensive if there are a lot of pixels very close together and the recursion function becomes very deep. A second method of connectivity grouping is to scan through the image row by row and if a pixel is found which hasn't yet been associated with a group, it is assigned a new group. Neighboring pixels are then assigned to that group as well. The image continues to be scanned linearly until there are no more adjacent pixels to group together. After all the pixels have been assigned groups, a second pass is made to connect adjacent groups. Groups which are touching one another are assigned to the lowest group number of those in contact. The effect is a parent child relationship between adjacent groups which is resolved until just a single group is left to represent each object. Grouping can be based on N4 or N8 connectivity. Figure 48 shows the connectivity grouping process.



Figure 48. Dual Pass Grouping by Connectivity

A new group is created for the first pixel in the first row and the second pixel is immediately associated with it. A second group is created for the first set of pixels in the second row. The second set of pixels in that row are associated with the first group since they are in contact with it. The same is true for the two pixels in the third row. In the forth row, the first pixel is associated with the second group due to connectivity, but the remaining pixels are assigned to a new group even though the last pixel connects to an existing group. The new group is defined since the first pixel is not touching an existing group. This new group is considered a child of the first group. The second pass is used to reconcile connecting groups with parent-child relationships and therefore the last group is absorbed into the first. This method has the advantage of only requiring a linear search of the data to assign the initial groups and a less intensive recursive search resolves adjacencies, thus reducing computation time. The disadvantage is that if there is a discontinuity to the object, the object will be split into to separate pieces.

7.3.2 Parametric Descriptor

The Parametric Descriptor is a method used to describe the shape of an object by characterizing it in some quantifiable fashion. A circle, for instance, can be characterized by its center-point and radius. In this way, all the pixel values representing the circle in the

image data can be reduced down to just two values and the image of the circle can still be redrawn if needed. This has the advantages of reducing the amount of data needed to represent an object in the image or being able to describe complex shapes within an image in a quantifiable way. This process is typically done after an object has been defined, either by extraction from the image or by a shape clustering technique. There are three basic means of parameterizing shape data, by a known equation, by measuring data points relative to a reference point, or by interdependent measurements, that is measurements relative to all the other points in the object data set. If it is known that there is a particular geometric shape present in the image, then it can be described by the corresponding equation and only the parameters of the equation be stored. This only works for basic shapes (lines, circles, rectangles, triangles, etc.) which can be describe using an equation. For general shapes, parameterization typically consists of measuring the offset of the data relative to another point in the image. Figure 49 shows two examples of this. One way is to measure the distance and angle from a pixel back to a common reference point. This reference point may be chosen arbitrarily, by a user, or through some mathematical means (such as calculating the centroid of the object). Distance and direction of all the data points comprising the object are then calculated and store relative to the reference point. This creates a descriptor then that can be used to recreate the object independent of the image from which is was taken. Another way to create a descriptor is to categorize the data relative to other data points of the object. This can be done relative to every other pixel or to only adjacent pixels. Again, this creates an independent representation of the generalized shape. Additional parameters can be included as well, such as scale, rotation, or projection. Adding parameters, however, increases the complexity of the descriptor which may be undesirable for additional processing steps.



Figure 49. Parametric Descriptors Based on a Central Reference Point and Interconnection

7.3.3 Robust Key Points

Robust key points are points in the data which contain a high level of descriptive information about the object they belong to [26], [27]. They may be the edges, an area of a unique texture, or a point which is present in all scales or transformations of the image. These are consistencies in the data which are maintained even with noise and image manipulation. Their robustness means that they are most likely to be present in all instances of the object and are therefore highly useful for categorizing the object. Typically, an object only has a few of these points, but since they are so unique to the object, only a few are needed to describe the object. These points are parameterized based on their consistent attributes and are store together to form a highly unique description of the object. The advantage of this is that is can be used to identify the object's presence in other image. The disadvantage is that the object itself is not stored, no can it be restored using the key points.

7.4 Object Detection and Identification

Object detection and identification is the key component of the analysis algorithm. There are several methods to detect target data within a given image. Detection, though, is only one part. Detection only states that there is an object found in the image. Identification states whether that object is what is being searched for. Identification tends to be a more difficult process which requires additional processing steps.

7.4.1 Sum of Absolute Differences

The sum of absolute differences is the simplest method of object detection. For each point in the frame image, an image of the target object subtracted and the resulting pixel values are summed and stored at that location. If the target object is present within the frame, then the pixel location with a value closest to zero, below some minimum detection threshold, is where the object is located. This method only works if the target object is of the same orientation and brightness as the matching object in the frame image. To detect variations of the object, a separate pass must be made for each permutation. This method will work for basic objects which are guaranteed not to change from one frame to the next in any way, other than their location.

7.4.2 Cross-Correlation

This method of object detection locates the point where the target object image is most similar to values within the frame image by performing 2-D correlation between the two data sets. For each point in the frame image, the pixels in the target image are element-wise multiplied with the corresponding pixels in the frame image and all the resultants are summed, normalized, and stored at that location. The target object is detected in the frame if the peak value of the cross-correlation image is greater than a set threshold. This method is more robust than the sum of absolute differences because it can still detect an object if lighting conditions have changed or the object is slightly rotated skewed. The limitations of this method is that it is exponentially more computationally intensive for larger image sizes. It also has issues when performing the cross-correlation at the edges where the target image is shifted such that it hangs off the bounds of the frame image. There are several ways to account this, including truncation, wrap around, or edge pixel duplication. None of these are perfect solutions.

7.4.3 Shape Number

The shape number, or chain code, detection method compares a descriptors of the shape boundary generated from the target image and the frame image. To create the boundary descriptors, edge detection first performed on the frame and the target image. The Canny method is used because the ideal boundary is a continuous, closed-loop, single pixel wide line. The descriptor, called the shape number or chain code, is formed by choosing an arbitrary edge pixel as a starting point and then locating the next adjacent edge pixel. The shape number can be computed using either N4 or N8 neighborhoods. A number is assigned to each possible pixel direction, 0-3 for N4 and 0-7 for N8. The number corresponding to the direction of the adjacent pixel is added to the chain. Figure 50 shows the generation of an N4 chain code.



Figure 50. Computing an N4 Shape Number (Chain Code)

Each connecting pixel in the chain is added until the original start point is reached or the edge segment ends. A 1-D cross-correlation is then performed between the target shape number and the frame shape number(s) to locate matches. Matching the descriptors using cross-correlation has two benefits. Since it is a similarity match, small perturbations in shape signature are smoothed out. Additionally, the starting point of the edge signature from the frame image can be arbitrarily chosen since the cross-correlation can cycle through all the values by using vector wrapping. The drawbacks of this method are that there is no way to create a rotation or transformation invariant descriptor and that accuracy is dependent on the quality of the edges extracted from the image data. A continuous, closed-contour edge provides the best descriptor, but this is not always possible to obtain. Discontinuous edges dramatically increase the likelihood of producing false or missing matches.
7.4.4 Hough Transformation

The Hough Transformation is an object detection method where each data point gets to "vote" on the likeliness that is part of a parameterized object descriptor [29]. This is typically used to identify lines or circles within an image; any object which can be described parametrically [30]. This method works by transforming each data point to a curve in the parameter space via a parametric equation or a lookup table of descriptor values. The parameter space acts as an accumulator for all the transformed data points. The local maxima, or areas where there are the greatest overlap of the curves, are then transformed back to the image space where they become the location of matches to the descriptor [31].

For geometric shapes, a parametric equation is used for the transformation. For example, a line is transformed to the parameter space via equation (7.7).

$$r = x\cos(\theta) + y\sin(\theta) \tag{7.7}$$

Non-geometric shapes require a descriptor to be generated since there is no basic transform equation. To create a descriptor, key points must first be extracted from the data. Typically is done by performing an edge detection on the image. A reference point is then defined from which to base the descriptor. This reference point may be the centroid, the spatial center, an edge point, or some other arbitrary point. For each key point located in the image a distance/angle pair is generated relative to the reference point using equation 7.8.

$$desc_{i}(d,\theta) = \begin{cases} d = \sqrt{(x - x_{c})^{2} + (y - y_{c})^{2}} \\ \theta = \tan^{-1}[(y - y_{c})/(x - x_{c})] \end{cases}$$
(7.8)

The distance (*d*) is the distance between the coordinate location of the key point (*x*, *y*) and the reference point (x_c, y_c). The angle (θ) is the angle, 0-360°, from the x-axis, of the vector line connecting the key point to the reference point. These values are shown in Figure 51.



Figure 51. Generalized Hough Transform Descriptor Parameters

The direction of the gradient is also computed for each key point and the descriptors are sorted based on ascending order of the gradient angle. This creates the final parametric descriptor for that shape.

Matching a target shape to a shape found in a new image is done by locating key points in the new image and using the shape descriptor to increment values in the parameter space. A peak in the parameter space above a set detection threshold indicates a match. To extract key points in the new image, the same method used to generate the key points for the shape descriptor is used. Again, this is typically an edge detection. For each key point found, the gradient angle is computed and the closest entry in the shape descriptor is selected. Using the direction/angle pair from that descriptor entry, the probable coordinates of the reference point (x_c, y_c) are computed using equation 7.9.

$$\begin{cases} x_c = x + d\cos(\theta) \\ y_c = y + d\sin(\theta) \end{cases}$$
(7.9)

The location $(x_{\sigma}y_{c})$ in the parameter space is then incremented by 1. The parameter space acts as an accumulator to tally the "votes" each key point casts based on the estimated location of the reference point of the shape. A peak in the parameter/accumulator space indicates a location where the most number of key points from the frame image agreed that the shape was present. If this peak value is greater than a set threshold for detection, then a positive match is made. There is a second method which can be used to approximate the Hough transform [29]. This method does not require that the image gradient angle be computed. By not computing the gradient angle, a decrease in processing time can be seen for simple matching within large images. Instead of computing and matching the gradient angle to the closest descriptor entry, all the descriptor entries are processed for each key point. Since the gradient angles are unused, they need not be computed when generating the descriptor either. The direction/angle pairs simply are added to the descriptor in the order they are computed. With all the descriptor entries added to the accumulator for all the key points, the accumulator values become much larger and are present within a larger area. This has the advantage of producing better object detection since there is more overlapping of the accumulated votes but has the disadvantage of producing worse object identification since the definition of the peaks are more vague.

The Hough transform method provides an accurate technique to located very specific irregular shapes within an image. Even the approximation of the transform is very effective at detecting and identifying complex shapes given relatively uncluttered data. The method also has the capability to be expanded to detect shapes independent of rotation and scale. This requires, though, additional parameters be added to the shape descriptor which increases dimensionality. Increased dimensionality increases processing time and resources required to perform a match.

8 PAW IDENTIFICATION ALGORITHM

8.1 Detection Requirements

There are many requirements imposed on the detection algorithm to ensure that it properly identifies the paws within the images without false or missed matches. The detection algorithm must be able to detect multiple, similar looking paws within a single frame, determine the position of the paws found, identify which paw has been located, and determine association between the same paw in previous or successive frames. The algorithm must also be robust to noise and variation in intensities within the image, variation in intensities from one image to the next, low contrast images, and extraneous objects in the view. Additionally, the algorithm must perform quickly and efficiently, as it will be required to process a large set of images in a timely manner.



Figure 52. Multiple Paw Contact Configurations

Within a single frame, there may be no paws or there may be all four paws. Typically, there are one or two paws on the surface with the others either coming or going. Figure 52 shows example of image with all four paws and with two paw striking during motion. Regardless of the number, any paw in contact with the surface must be registered. Once located within a frame, each identified paw must be associated with its corresponding instance in previous and successive frames. This must be spatially localized though, as there will be multiple strikes of the same paw, and similar appearing paws, at differing locations within the video. This measures its persistence in time throughout the data set.



Figure 53. Flaws in the Image Data

The algorithm must be able to process frames which may be corrupted by noise, non-uniform intensities, or extraneous objects, as shown in Figure 53. Noise can arise from over-amplifying the gain added to the image by the camera, video compression, or low quality overhead florescent lighting. The algorithm needs to be able to filter this out without losing useful image data. Non-uniformity in intensity can occur within a frame, from one region to another, or between frames, in which the median intensity shifts. Most non-uniformities are removed or reduced via the optimal lighting scheme of the physical setup, however, even under the best conditions, there still remains some variation in intensity. The algorithm must be advanced enough to correct for non-uniformities or avoid methods which would be sensitive to such variations.

Target detection and target identification are both required of the algorithm. Though similar, target detection is used to determine if a type of object is present within an image, target identification matches the object to a specific template and returns the location within the frame. For basic detection, the target object can have very loose bounds on it definition. For example, it may be a line or a circle, though the angle of the line or the radius of the circle is not defined. Detection will simply determine if the object type is present anywhere in the frame. Target identification will match a circle of a specific radius and report its location within the frame at the exclusion of other circles not matching the template radius. To apply detection and identification to paw matching, detection would determine if a frame contained a paw or multiple paws. Identification would assign value to each paw found (e.g. fore or

hind) and specify where in the frame each paw was found. Identification, however, increases computational complexity because the algorithm is searching for a higher precision match. Multiple similar objects, such as four mouse paws, can be easily detected, but require greater processing to properly identify.

8.2 Implementation

There were two implementations of the paw detection and identification algorithm. The first one was based on matching a target descriptor to one extracted from the image data. The descriptors were generated by measuring the distance and angle between the centroid of a region thought to be a paw and the centroid of the smaller pixels groups within the region. These pixel groups were found using the difference of Gaussian method. This method proved to have too many false and missed detections, so a second algorithm was developed. The second algorithm uses the Hough transform to match a descriptor of edge points from a target image to the edges found in the frame image. When a paw is detected, it is registered to the region of the image it was found and a new target descriptor is generated for use in subsequent frames. This method provides higher accuracy for positive identification.

8.2.1 Descriptor Matching Based on Centroid of Difference of Gaussian Regions

This implementation of the paw identification algorithm is a multi-step process which uses a variety of image processing techniques to extract and categorize relevant image data, that is, areas in the image where there are paws in contact with the plate. This method processes a single frame at a time, independent of other frames in the sequence. The first step is to extract all the objects in contact with the plate from the rest of the frame image. To do this, a multi-level, difference of gaussian image set is computed from the frame. The initial sigma value used is $\sigma = \sqrt{2}$. It is decided that the second level of this set provides a good representation of the objects in contact with the plate, while minimizing background noise (See Figure 46 in Chapter 7). To reduce computation time, only the desired level can be computed. For the second level, this corresponds to sigma values of 2 and 4. The pixels in the raw DoG image are then placed into groups based on N–4 connectivity. This is a dualpass process where the pixels are first assigned a group and then adjoining groups are merged. Groups less than a set size (2x2) are considered extraneous and are removed from the data. Each pixel is then reassigned a value corresponding to its group number. The pixel groups are then assigned to a larger collection of groups based on proximity. This is a recursive process where a group is added to a collection if it is within a set distance (15 pixels) of another region. Figure 54 shows the results of this process on an image frame. Figure 54B is the Difference of Gaussian after a black and white threshold has been applied. Image C shows the result of pixel grouping. The gray value of each pixel corresponds to the group number it belongs to. Figure 54D shows the results of grouping nearby pixel clusters into complete regions.



Figure 54. Pixel Clustering of DoG Data

For each collection of pixel groups, a descriptor is generated based on the distance and angle from the centroid of the collection to the centroid of each of the pixel groups. The descriptor is a 120 element vector where each element represents 3° of rotation. Each distance and angle measure is decomposed into the closest two angle bins and the resulting distances are added the corresponding vector elements. The resulting descriptors are then compared with a master target descriptor of each paw. The master target descriptors are computed using the same process from an ideal paw image. The matching is done by finding the descriptors

which have the smallest "distance" from the target descriptor. Equation 8.1 is used to compute the closeness-of-match distance for each element shift of the image descriptors.

$$Dist(shift) = \sqrt{\sum_{i=1:end} [tdesc(i) - pdesc(i + shift)]^2}$$
(8.1)

The shifting is done to match all rotations. The values are wrapped as they are shifted until a complete rotation is tested. A small distance between the two descriptor means they are closely matched. A matched descriptor will represent a paw in the image. It is possible for multiple paws to be located in a single frame. The center point of the paw is given by the centroid of the collection of pixel groups. The rotation of the paw can be found to the nearest 3° and is given by the value of the shift parameter where the best match was found.

The algorithm is able to detect paws in the image with ~70% accuracy. This can be increased to ~85% accuracy but the number of false detections increases as well. Also, the algorithm is not able to discern one paw from another. Only paw detection within a frame is possible, not identification. False detections are sometimes registered when the animal's face is close to the plate. The algorithm in unable to detect the paws of black mice. This is because the image data for black mice is very high contrast compared to white mice. The difference of gaussian therefore no longer provides multiple clusters of pixel for each paw. Instead the paws are seen one large shape. This gives the corresponding descriptor zero elements to match with. It was ultimately decided that this method of detection was flawed in its designed since it relied on imperfect data to generate the match descriptors. Further development was subsequently abandoned.

8.2.2 Locally Adaptive Generalized Hough Transform

This paw identification algorithm utilizes edge detection and the generalized Hough transform to locate paws in the frame. It is optimized for use with the camera/plate data acquisition system. This method processes a single frame at a time, but the search areas are dependent on each paw location from the previous frame. The background of each frame in the sequence is normalized to provide better paw contrast in each frame, remove extraneous

objects or imperfections on the plate, and reduce noise. To normalize the background, the threshold separation technique discussed in section 6.3.1 is performed. The first frame of the sequence is used to create a uniformity mask. The mask is then added to each of the frames in the sequence. The caveat of using the first frame to generate the mask is that the frame must be empty. Figure 55 shows the uniformity mask (A) for a frame sequence and the results of the intensity normalization (C) on an image frame (B).



Figure 55. Uniformity Mask, Raw Frame, and Background Normalized Frame

Next, edge detection is performed to extract all the edges in the sequence. This can be done across the entire sequence at once, or can be performed just before the target matching step. Performing edge detection just before target matching has the advantage of only requiring a single frame of edge data to be stored rather than a copy of the entire sequence being held in memory. For the white mice, the zero-cross method of edge detection is used. This is performed using the first Laplacian operator shown in 6.L with a threshold of 0.002. This highlights the low contrast outer edges as well as the texture within the paw while removing the greatest amount of noise. For black mice, the canny edge detection method is used. This is performed with a sigma value of 2 and thresholds of 0.3 and 0.12 (40% strong edge threshold). For black mice, the canny method outlines the paws very precisely. The canny method is the preferred method as it does an excellent job of outlining just the paws in contact with the plate. The white mice data is too low contrast to properly use this method, so the zero-cross method is used instead to obtain a good approximation of the edges with minimal noise.

A master target descriptor is predefined for each of the paws. The master image is selected by comparing numerous images of the same paw and choosing one which best defines the canonical shape. Figure 56 shows the comparison of images of the back paws of a white mouse used to select a master target descriptor.



Figure 56. Left and Right Hind Paws Samples Used to Generate Hind Paw Descriptors

The descriptor is generated by first performing a canny edge detection on the master paw image. The edge lines are then cleaned up by hand to produce a prototypical representation of the paw. The image is cropped to contain only the edge pixels and a center point is chosen as reference point to based the descriptor around. The final result is shown in Figure 57.



Figure 57. Master Target Descriptor Image for White Mouse Hind Right Paw

The distances and angles from the center reference point to each of the edge pixels are then computed. These distance/angle pairs form the final parametric descriptor. It was found that for improved detection, each paw would have multiple descriptors representing various poses. Each paw, therefore, has three descriptor sets, as seen in Figure 58, one for the average paw shape, one for the paw with the digits spread far apart, and one for the paw with the digits pinched close together. There are also separate descriptor sets for white mice and black mice. For the front paws, there are an average of 174 distance/angle pairs per descriptor. For the back paws there are an average of 341.



Figure 58. Descriptors Based on Variation in Paw Poses

Target matching is implemented by performing a Hough transformation of the edge enhanced image for each of the paw descriptors. This process starts by first searching the edge enhanced frame image for edge points. When an edge point is found, values in the accumulator space are incremented (or "voted" on) at the locations computed by the distance/angle pairs of the paw descriptor centered about the edge point. The result in the accumulator space after all the edge points have been processed is a peak where the center point of the paw is suspected to be. The peak represents the location where the greatest number of edge pixels agree that the reference point, which the paw descriptor is based on, lies. If the peak is above a set detection threshold, then the location is considered to be a positive match. However, there is rarely ever a single peak, but rather a collection of smaller, highly localized peaks. Therefore, for greater accuracy, its better in practice to smooth out the peaks. This is done in two ways. First, for each vote in the accumulator, the pixel value at the location computed by the distance/angle pair is incremented by 3 and the surrounding pixels are each incremented by 1. Secondly, after all the votes are cast, the entire accumulator is smoothed by convolving it with a 5x5 identity matrix. The smoothing removes jagged areas around the peak locations and accentuates the peaks themselves. This makes it easier to find a single local maximum rather than several smaller maxima grouped closely together.

The Hough transform is able to both detect and identify each paw, including multiple paws within a frame with a high level of accuracy. There are instances though, particularly when processing the low contrast white mouse frames, when one paw is falsely identified as another. A left paw, for example, may be mistaken as a right paw. When the back paws lift up from the plate, they can appear very similar to a front paw with the digits spread out. To reduce the number of inaccurate matches and to reduce computational complexity, a bounding system is implemented to track and predict the positions of the paws as they transition through the sequence of frames.

Each paw is assigned its own bounding box with values for the leading, trailing, left, and right edges. Initially, the bounds on the left paws are set to the full frame length and to the 60% of the image width from the left side. Likewise, the right paw bounds are set similarly, but with the width range justified to the right side. This provides a cross-over area of 20% image width in the center where both left and right paws may be detected. The length bounds are initialized to the full length of the image so that if a paw misses detection early in the sequence, it can still be detected in subsequent frames. When a paw is detected, the bounds are immediately adjusted to window in on the paw. For front paws, the bounds are set to the width of the paw plus 20 pixels on each side and the length of the paw plus 30 pixels on each

side. For back paws, the bounds are set to the width of the paw plus 35 pixels on each side and the length of the paw plus 50 pixels on each side. The buffer space is added to ensure that no edge pixels belonging to the paw are missed by the Hough transform as the transform is only performed within the bounding area of each paw. When the front paw bounds are updated, the back paw bounds on the same side are updated as well. Since back paw will never lead the front paw, the leading edge of the back paw can be set to the trailing edge of the front paw. Once a detected paw ceases contact with the plate, the bounds are reassessed to predict the path of the animal. The leading edge is extended to 50% of the total frame length past the last detected center point. For a left paw, the left edge is reset back the far left side of the frame and the right edge is set to half way between the current right edge value and the current left edge value of the right paw. For a right paw, the same updates are made only with the right and left side edges switched.

Dynamic windowing results in a number of advantages when using the Hough transform. It reduces computation time because the transform is only applied to the points within the window. It forces the algorithm to only look for a particular paw within a predicted area, thereby reducing false matches. Lastly, it reduces the number of misses since the detection threshold within the window can be decreased without resulting in an increase in false matches.

To match paws within each frame, the accumulator spaces are searched for the peak values which are larger than the detection threshold. The location and strength of peaks found within the detection boundaries are stored and the maximum strength peak from a complete paw descriptor set is compared to the detection threshold for that paw. If the maximum strength peak is greater than the threshold, then it is considered a match and the location of the peak is stored as the center point of the paw. For each frame, the center points of all four paws are logged. If a paw is not present in the frame, it's center is logged as (-1,-1). If a new match is found, then, in addition to adjusting the bounding region, a new dynamic descriptor is generated based on the edge data from the frame. The distance/angle pairs for the new adaptive descriptor are computed for each edge pixel within the bounding area relative to the

center point given by the Hough transform match. This new descriptor is then added to the set of descriptors for that paw to be used for matching in the next frame. The dynamic descriptor is updated each time a positive match is made. By matching to a dynamically changing descriptor, the paw can be tracked with greater accuracy through subsequent frames. Without using the adaptive descriptor, the strength of the maximum peak falls below the detection threshold shortly after the initial match. This is because the master descriptors are based on an image of a firmly planted paw. As the paw shifts off the plate, the match strength of the master descriptors drops. The adaptive descriptor, there are independent adjustable detection thresholds for each paw. The thresholds are initialized to a set value (White mice: Front = 37, Back = 35; Black mice: Front = 41, Back = 37), but for each frame that a match is found, the value is reduced by 2. The threshold is then reset when the paw ceases contact. This also helps to maintain paw tracking after the initial match is made.

The final result of the frame processing is a list of all the paws identified. The list is indexed by frame, and for each frame, there are four entries, one for each paw. These entries contain the location of the paw within that particular frame. The next step is to associate occurrences of the same paw from one frame to the next. A new list is created to organize the gait data by order of footfall. The raw gait data is searched frame by frame, and for each new instance of paw contact, a new entry is made which stores the average center point and the frame numbers where contact begins and ends.

Finally, to ensure that the same paw is not identified twice in the same frame, the gait data is searched for center points with close proximity during the same time span. There can be occurrences when a back paw is registered as both a back paw and a front paw. This typically happens near the end of the run when the front paws are out of frame and the back paws are lifting off the plate. It is assumed then, for these cases, that the doubly identified paw is truly a back paw. If such a case is found in the gait data, then the correspond front paw data is removed from the set.

This method of detection requires a lot of fine tuning of the threshold values and boundary conditions, but once all values are set, it provides a highly accurate method of both detecting and identifying paws through a sequence of images. This method is used in the final implementation of the gait analysis system. The experimental results garnered from this method will be presented in the following chapter.

8.3 Final Implementation

The final implementation of the detection and identification algorithm is based on the locally adaptive generalized Hough transform method. This method was chosen over the Difference of Gaussian based method because if proved to be more robust to image noise, non-uniform background intensities and extraneous. The Hough transform method also uses more well defined descriptors. The results of the final implementation of the algorithm will be presented in the following chapter.

The final algorithm uses the approximation method of computing the Hough transform. The approximation is used because it is simpler to implement and requires less memory resources than the full Hough transform implementation. The disadvantage of using the approximation is that paws occasionally miss detection. To correct for this, descriptor sets of multiple poses of each paw are used to increase the likelihood of detection. Although using multiple pose descriptors increases the identification accuracy, it also increase the run-time of the algorithm. The time to process an average gait data sequence, approximately 100 frames, is 520 seconds, or 5 seconds per frame. This time can vary though. Frames where paws are already in contact with the plate will process faster since the paw location is already known and wide-range search need not be conducted.

There are two major limitations to the identification algorithm. It requires the mice to continually make forward progress and it is not invariant to paw rotation. For identification to occur, no paw may be placed behind its last known location. If the animal stops part way through a run and steps backwards, the paw contact point will not be located. This is due to the predictive nature of the algorithm. The algorithm is set to only look for paws which

correspond to proper forward moving animal gait. The second limitation is that the identification process is not invariant to paw rotation. Paws can still be identified within $\pm 2^{\circ}$ of rotation relative to the paw descriptor. The algorithm can be made rotation invariant but this requires that descriptors be generated for all possible rotations. Matching to a greater number of descriptors, while possible, increases the run-time to unacceptable limits.

9 EXPERIMENTAL RESULTS

9.1 Introduction

This chapter presents the results of the final design of the gait acquisition system. The results shown here are gathered from experiments conducted on live animals using the Fleet Foot - Mark II system and the Locally Adaptive Generalized Hough Transform identification algorithm. Data presented includes the raw image data acquired from the system and the data reported to the user based on the image data. A performance analysis of the system is also detailed.

9.2 Live Animal Experiments

Live animal experiments are conducted in a four step process: 1.) Configuring the device for the shade of mouse, 2.) Loading the animal, 3.) Activating the system for capture and opening the walkway doors, 4.) Removing the animal and cleaning the device surfaces. The device setup differs slightly for white mice and dark (black or brown) mice. For a white mouse experiment, the dual level lighting scheme is used and the walkway hood is set such that the white side is facing down toward the plate and camera. For a black mouse experiment, the dual level lighting is not needed, nor desired, and therefore the top level of lights (above the plate) are disconnected. The walkway hood is set such that the black side is facing the plate and camera. An experiment is started by loading a mouse into the light side house, closing the top, and setting the gate in place to separate the house from the walkway. The PC software is then used to activate the trip sensors, initialize the camera for capture, and prepare the computer for data recording. When the trip sensor light, this indicates that the system is primed and in standby for image capture. The gate is then removed to allow the mouse onto the walkway. As the mouse crosses the walkway, the sensors are tripped and the PC software begins data recording. When the mouse reaches the dark end, the second set of trip sensors are triggered and the recording ends. The gate is also inserted back between the walkway and the dark house to keep the mouse from crossing

back over. The mouse is then removed, the houses and walkway are cleaned of any contaminants, and the gate is place back between the light side house and the walkway. The system is now ready for another run. After all experimental data is collected, it can then be analyzed independent of the physical device. The live animal experiments with the final design have yielded usable mouse gait data. The experiments document the progression of recovery after an injury and prove the feasibility of the system for tracking spinal injury recovery.

9.2.1 Healthy White Mice

Figure 59 shows White Mouse Aqua-1. This image was taken from data of the first run captured on 8-2-05. This was a pre-operation run conducted to collect healthy animal gait data. For this experiment, the animal's paws were not inked as they had been in the initial experiments with the first Fleet Foot design. Inking had proved to have too many disadvantages, so instead, this series of runs (and all subsequent experiments) was captured using the newly designed dual level lighting scheme. Using the dual level lighting, the background and body merge together very well and the paws remain in comparatively high contrast. The paw identification markers present in the image are placed on the computed center point of each paw. The different shapes of the markers identify which paw was located; X for fore-right, + for fore-left, - for hind-right, \diamond for hind-left.



Figure 59. Healthy White Mouse with Paw Identification Markers

Figure 60 shows the trace of White Mouse Aqua-1's gait. The chart displays paw contact for each frame as an on/off value where the presence of graph bar represents paw contact. The

chart is organized with the forelimbs as the center two traces and the hindlimbs as the outer two traces. The strike pattern of healthy interlimb coordination follows fore-right \rightarrow hind-left \rightarrow fore-left \rightarrow hind-right and then repeats. During coordinated stepping, opposite, nonadjacent paws strike in quick succession with the hind paw lagging slightly (e.g. fore-right, hind-left will strike within 1-2 frames of one another). This is followed by a contact/swing time in which locomotion occurs. The first two paws then raise from contact and the opposite two quickly strike to maintain balance. Full coordinated stepping is shown to be present in Figure 60. Starting with the first fore-right paw strike, the gait pattern follows each of the healthy gait criteria over a total of 9 paw strikes. This represents two complete strides with a third being composed of the leading portion of one at the start of the graph and the trailing portion of another at the end of the graph.



Figure 60. Gait Trace of Healthy White Mouse

9.2.2 Injured White Mice

Figure 61 shows White Mouse Blue-2 post spinal trauma. This image was taken from data of the third run captured on 7-20-05, four weeks post-injury. It is shown that the hind limbs are entirely incapacitated, as they are palm up and dragging behind the animal. This implies that the front paws are supporting all the weight and forward locomotion.



Figure 61. Injured White Mouse with Paw Identification Markers

Figure 62 shows the trace of White Mouse Blue-2's gait. The graph supports the assumption that all the weight support and locomotion is produced by the forelimbs. The quick, short burst of movement followed by long periods of contact and virtually no hindlimb contact displays typical injured gait behavior. The brief spikes in the hind limb traces are due to false detections arising from matching the dragging paws.



Figure 62. Gait Trace of Injured White Mouse

9.2.3 Healthy Black Mice

Figure 63 shows Black Mouse Red-2. This image was taken from data of the second run captured on 8-24-05. This was a pre-operation experiment conducted to collect healthy animal gait data. The black mice, as can be seen, have paws with much greater contrast to the body and the background than that of the white mice. Because of the improved contrast, the lights above the walkway are not needed and are disconnected for all black mice experiments. Additionally, the hood is set so that the black side is facing the camera, thereby providing a black background.



Figure 63. Healthy Black Mouse with Paw Identification Markers

Figure 64 shows the trace of Black Mouse Red-2's gait. The healthy interlimb coordination pattern is present for the first half of the sequence and three complete strides can be distinguished from the data. The second half of the data indicates that the animal stopped just prior to triggering the second trip beam. This is common among the black mice. They are either more curious about the red light of trip beam or more wary of the dark mouse house or both. It remains to be seen if better training will remove this delay. It should also be noted that during the coordinated stepping, opposite non-adjacent paws make contact nearly simultaneously as their counterparts are lifting from the plate. This differs from the results seen in the white mice where the fore paws typically lead the hind paws.



Figure 64. Gait Trace of Healthy Black Mouse

9.2.4 Injured Black Mice

Figure 65 shows Black Mouse Red-2 post spinal trauma. This image was taken from data of the third run captured on 9-16-05. Spinal trauma has resulted in the complete loss of hind limb movement.



Figure 65. Injured Black Mouse with Paw Identification Markers

Figure 66 shows the trace of Black Mouse Red-2's gait. There is no interlimb coordination present as the hind limbs have been completely disabled. The injury gait pattern of short, quick fore paw steps is seen instead.



Figure 66. Gait Trace of Injured Black Mouse

9.3 Hit/Miss/False Detection Ratio

The Hit/Miss/False (H:M:F) detection ratio is the percentage ratio of positive paw identifications (hits), to missed paw identifications (misses), to false identifications (false detects). A positive paw identification is defined as any point where a paw is located and the identification algorithm correctly detects its presence and classifies it. The goal of the algorithm is to generate 100% positive matches.

A missed paw identification is defined as any point where a paw is located and the algorithm fails to detect it. The result of this is a gap in the gait data. There are several causes for missed detections. The primary cause is paw rotation. Since the algorithm is not rotationally invariant, paws which are rotated past the tolerance of the descriptors fail to be detected. Low contrast regions are the second main cause of missed detections. These occur when the paw is tucked under the body or placed against the background in such a way that intensity contrast significantly drops. When this occurs, edges are not properly located and there is not enough data to compare with the descriptors to raise the match value above the detection threshold. The third main cause of missed detections is out-of-bound errors. This error occurs when the paw is placed outside of the prediction window. This often happens when the mouse stops mid run and then continues again in another direction and crosses a paw over the previously computed centerline.

A false paw identification is defined as any point where a paw is located and the algorithm correctly detects its presence, but falsely classifies it. This can happen when a paw crosses to far past the centerline and is identified as the opposite side paw or when a hind paw is lifting off the plate. As the hind paw lifts up it appears similar to a front paw. False detections can also occur near the dark side edge capture area. In this region, both fore and hind paws can be detected in the same location. It is equally likely that a front paw may make contact just within the capture area as it might at some point just outside. This depends on how the animal is lined up when it enters the capture area. If the front paw contacts the plate out of frame, then the hind paw will make contact in the edge region. Since the algorithm is searching for both in the same location, a hind paw can be confused for a front paw. The algorithm performs a search for dual detection at the same location, but there remain some occurrences when a hind paw is falsely identified at a front paw.

For healthy white mice, the H:M:F ratio was found to be 87.3% Hits : 10.4% Misses : 2.3% False Hits. For healthy black mice, the H:M:F ratio was found to be 86% Hits : 11% Misses : 3% False Hits. The ratio for the white mice was computed from 21 separate runs. The ratio for the black mice was computed from 33 separate runs. For the white mice, 76% of missed detections were due to low contrast images leading to poor edge data. All the false hits were due to previously missed paws being incorrectly identified in later frames because the prediction windows were not updated. It was also found, particularly for the black mice, that paws which missed detection initially would be detected a few frames later after the animal had shifted its weight and rotated the paw just slightly enough to bring it into the rotational tolerance of the descriptor. For the black mice, 43% of the missed detections were identified

at hits in subsequent frames. Also, for black mice, 53% of the total number of misses were caused by rotation outside of the detection range. To correct for these misses, additional descriptors can be added to account for slightly more inward and outward paw rotation. This has the drawback however of increasing computation time. It should be noted though, that for experiments where the animal proceeded straight across the walkway, without hesitation or variance, there was 100% positive identification.

9.4 Mice Training

Testing has proven that it is extremely important to train the mice to run across the device before conducting real experiments. This ensures that the animal is comfortable with the system and knows what is expected of it. Untrained mice are hesitant and often stop part way through the run, resulting in unusable gait data. Not only do trained mice yield better data, but it is less time consuming to conduct the experiment when the animal is cooperative. There have been several successful training methods used. One method is to place the mouse in the device with access to both houses and let the animal explore the area for 10-15 minutes. This allows the mouse to explore the area and become comfortable with device. Another method is to provide a food incentive to the mouse at the dark end of the device. This trains the animal to run across the walkway for a reward. A third method is to place the mouse in the device and then gently push it out of the houses onto the walkway every time it enters one. This trains it to leave the house as soon as it enters one. If repeated daily a week prior to the experimental run, these training methods can result in greatly improved gait data by minimizing idle time and exploration on the walkway.

10 CONCLUSION

10.1 Introduction

This chapter present a analysis of the final data acquired from the gait acquisition and analysis system. First, the raw data reported to the user from the detection algorithm is given. The gait metrics derived from this data are then presented and explained. Next, a description of the layout and use of the gait trace charts is detailed. This is followed by a detailed summary of an entire injury experiment where the level of recovery is ascertained on a weekly basis using data provide by the system. Next, a comparison between the system and the previous method of evaluating recovery, the BMS score is presented. This is followed by a final discussion on system design alternatives for future study. The chapter is concluded with an overview of the entire project.

10.2 Gait Data Reported

The gait data generated by the identification algorithm is reported to the user via an Excel spreadsheet. Table 3 is an example set of pre-op gait data from white mouse Aqua-1.

Paw	Center-X (cm)	Center-Y (cm)	Start Frame	Stop Frame	Strike Order
FLF	17.7	1.8	9	17	FLF
FLF	11.0	1.7	25	32	FRF
FLF	4.3	1.5	39	49	BLF
FRF	14.4	3.2	17	25	FLF
FRF	7.6	3.1	32	40	BRF
FRF	1.3	3.1	49	59	FRF
BLF	18.4	1.8	19	25	BLF
BLF	11.6	1.5	34	44	FLF
BLF	5.0	1.3	50	64	BRF
BRF	14.6	3.1	27	35	FRF
BRF	7.9	3.0	41	52	BLF
BRF	1.9	2.9	60	68	BRF

Table 3. Raw Gait Data Generated by the Identification Algorithm

The data is organized by paw (front left, front right, back left, back right). The x and y coordinates of the paw center points are reported in centimeters, based upon the upper left corner of the frame. The frame number where contact started and contact ceased for each paw strike is given. Finally, the order of paw strikes is shown separately. In addition to this data, the identification algorithm generates a second data sheet which provides the contact and swing durations of each paw. This data is used to generate the gait trace graphs.

10.3 Analysis of Gait Data

From the data given by the identification algorithm, a number of desirable gait metrics can be obtained. The center point coordinates allow for any spatial measurement to be computed, the most relevant being stride length and stance width. The frame numbers provide timing data which can be used to derive stance duration, swing duration, and velocity. Interlimb coordination can be measured by observing strike order and contact time.

	Average Stride	Average Stride	Average Contact	Average Swing
	Length (cm)	Time (msec)	Duration (msec)	Duration (msec)
Front Left	6.5	258	139	125
Front Right	6.5	275	144	133
Back Left	6.7	258	167	125
Back Right	6.4	275	150	117
Average Stanc	e Width (cm)		Coordination	
Fore	1.48		Number	
Hind	1.45		45	

Table 4. Computed Gait Metrics Based on Acquired Data

The metrics derived from the data set in 10.1 for mouse Aqua-1 are shown in Table 4. The *Average Stride Length* is the direction-of-motion distance from one paw strike to the next of the same paw, measured in centimeters. This is found, for each paw, by averaging the differences of the x-coordinates between each paw strike. The *Average Stride Time* measures the amount of time from the beginning of one paw strike to the beginning of the next of the same paw, reported in milliseconds. This value is computed for each paw by calculating the difference of the start frame number between each strike, dividing by the capture frame rate

(60fps for this data set), and then averaging the results. The *Average Contact Duration* is the average time the paw spends in contact with the plate, measured in milliseconds. Contact duration is derived by computing the difference between the start and end frame numbers for each paw strike. The *Average Swing Duration* is the average time the paw spends between strikes. Swing duration is derived by computing the difference between the difference between the end frame of one paw strike and the start frame of the next paw strike of the same paw. With all time data, the resolution of the calculated value is dependent on the capture frame rate. For this data set, with a frame rate of 60fps, the time is measured in increments of 0.0167 seconds. The *Average Stance Width* is the across-the-body distance between the right and left paws, measured in centimeters. The stance width is found, for fore and hind paws, by computing the difference between the left and right side paw y-coordinates for the same strike number.

The *Coordination Number* is a score devised to rate the level of interlimb coordination of the mouse. This number is based on the strike order and the healthy coordination pattern (fore-right \rightarrow hind-left \rightarrow fore-left \rightarrow hind-right). The coordination number is computed by assigning an incrementing value to each strike transition which follows the healthy gait pattern. The values are then summed to get the final coordination number. If there is a break in the gait pattern, the value assignment resets. A pattern of hind-left \rightarrow fore-left \rightarrow hind-right \rightarrow fore-right \rightarrow hind-left would be assigned the values [1 2 3 4] for the four healthy gait transitions. The coordination number is then 1+2+3+4 = 10. A pattern of fore-left \rightarrow fore-right \rightarrow hind-right \rightarrow fore-right \rightarrow hind-left would be assigned to this sequence [0 1 0 1 2] for a coordination number of 4. For the Aqua-1 data set, there are 9 successive paw strikes which follow the healthy gait pattern. The values assigned to this sequence are [1:9] which gives the final coordination number of 45. If there was a break in the pattern half way though the strike sequence, the new values would be [1 2 3 4 0 1 2 3 4] to give a coordination number of 20.

Gait trace charts provide a quick visual method to convey acquired gait timing data and strike order. This allows for a cursory assessment of the recovery state of the mouse based on quantitative measures. The gait traces shown in Figure 67 were selected randomly from healthy mouse gait data sets where the mice ran directly across the plate. Interlimb coordination can be visually identified as the checkerboard pattern in the graphs. This represents the alternating strikes of pairs of opposite non-adjacent paws. The upper-left to lower-right diagonal striping of the contact start frames is also indicative of interlimb coordination. The two full strides and partial third stride are apparent in the data as well. The data capture ends at the trace with the highest frame number. Variations in the length and spacing of the traces correspond to animal behavior during the capture and to subtle gait differences intrinsic to each animal strain.



Figure 67. Healthy Animal Gait Traces Showing Interlimb Coordination

The animal behaviors are easy to recognize from the data. The subtle differences in gait are more difficult to catagorize as there is no preexisting, comprehensive, mouse gait model, for each individual strain, to compare to. The two mouse behaviors most distinguished in the data are idle time and average velocity. Idle time is when the mouse has entered the capture area but severely slows or complete ceases motion. For healthy mice, there are typically two reasons for idle time: the mouse stops to examine the walkway, or something found on it, or it becomes wary of the dark end house or second trip beam and pauses just before triggering the capture to end. This behavior can be seen particularly well in the final elongated traces in Figures 67 C and D. In Figure 67C, the uniform length and spacing of the traces in the first 35 frames indicate the mouse was moving with a consistent steady gait. In the latter 35 frames, the hind-left, fore-left, and hind-right limbs maintain contact with the plate for nearly the full remainder of the capture time. This indicates that the mouse paused just prior to triggering the second trip beam. The raise in the fore-left and hind-right paws just before capture ends, as well as the final short step taken with the hind-left paw, shows where the mouse continued forward movement and progressed far enough to stop the data capture. This behavior is supported by the video record.

General velocity is the second prominent animal behavior distinguishable from the gait trace charts. There are the same number of traces in Figures 67A and B, yet a quick inspection of the charts will show that mouse Blue-2 in graph B was moving faster than mouse Aqua-1 in graph A. The traces in graph A begin in frame 9 and end in frame 68. The traces in graph B begin in frame 11 and end in frame 49. Although there is only a 2 frame difference between the start times, the end times differ by 19 frames. The length of the traces in graph B are approximately 8 frames long whereas the length of the traces in graph B are approximately 6 frames long. The overall capture time and the length of traces both indicate that mouse Blue-2 traversed the same walkway distance in less time than mouse Aqua-1, and therefore was moving at a greater velocity. The data supports this, showing that the average contact time of Blue-2 is 6.08 (101msec at 60fps) frames and its velocity is 24.5 cm/s. The average contact time of Aqua-1 is 9.0 frames (150msec at 60fps) and its velocity is 17.7 cm/s.

In additional to indicating animal behavior, the gait trace charts display subtle difference between the gait characteristics of different strains of mice. The most perceptible difference being the timing of the hind limb strike when a new interlimb pair makes plate contact. In white mice, there is a 1-2 frame lag between the forelimb contact and the alternate hindlimb contact. This delay is not as prevalent in black mice and can be, at times, reversed, with the hind paw being detected a frame before the fore paw. Without a standard strain-by-strain models of mouse gait to compare to, these nuances, and their effect on the overall gait are overlooked.

10.4 Tracking Recovery Using Gait Data

Recovery of an injured mouse is measured as the progressive reestablishment of the healthy gait pattern. Interlimb coordination, stride length, stance width, stance duration, and swing duration will change based on the level of injury and phase of recovery. To show how the recovery of an injured mouse can be tracked, the gait data from a single mouse acquired over a several week span is presented. The injury is a directed, quick impact, spinal cord trauma with the intention of disabling the animal's hind limbs. The spinal cord injury was successfully administered to black mouse Red-2 resulting in complete loss of hind limb functionality. Figure 68 shows the gait traces of mouse Red-2 taken before the spinal cord injury and then once a week for two weeks post-operation. Graph A shows a healthy interlimb coordination pattern and has be discussed in the previous section. Graph B shows the gait trace of the animal one week post-op. It is clear from the trace that the hind limbs are no longer supporting weight or contributing to forward movement. The hind limbs, completely incapacitated and dragging behind the animal, are no longer supporting weight or contributing to forward movement and result in no trace on the chart. The forelimbs are providing all locomotive drive through a series of exaggerated, plantar steps punctuated by quick, staccato movements. The short swing duration of the forelimb steps indicate that all weight support is maintained by front paws. The longer contact times indicate that all forward motion is being driven by the front paws as well. The video supports the interpretation of the chart data, showing quick bursts of paw movement during repositioning followed by long labored movement as the mouse drags the rest of its body along. It has also

been shown the injured mice tend to vary less from a straight path as their hindered movement makes them less inclined to explore.



Figure 68. Gait Traces Showing Recovery of Black Mouse Red-2

Figure 68C shows the gait data acquired two weeks after spinal cord injury. The chart shows partial recovery of hind limb motility. The hind-right trace indicates that motor skill has been partially restored to the paw. The hind-left paw is still dragging behind however as its trace is absent from the graph. The forelimbs, while still providing the majority of forward drive, as shown by the long contact times, are no longer as quick and numerous as they were the previous week. This indicates that the hind-right limb may be contributing some weight support and/or propulsive force. Interlimb coordination between the fore-left and hind-right paws is also starting to emerge again. The strike order is correct, but only one sequence occurs timely enough to effect the coordination number, giving this run score of 3. Upon review of video, it is evident that there has been some motility restore to the hind-left paw as well, but not enough to make plantar contact, support weight, or aid in forward locomotion. Figure 69 is a frame taken from the video showing the plantar stepping of the hind-right paw and the instability that remains in the hind-left paw.



Figure 69. BM Red-2 After Partial Injury Recovery

10.5 Comparison to BMS Score

The goal of this research is to have developed a system that can, by gathering qualitative data pertaining to mouse gait, augment or even substitute the BMS score (Basso Mouse Scale), currently the defacto method of assessing spinal cord trauma recovery in mice. As discussed in the introductory chapter, there are three phases of recovery associated with the BMS score, each pertaining to a marked change animal health. The first, or early recovery

phase, covers limb movement with no weight support and scores as follows: 0 for no movement, 1 for slight ankle movement, 2 for extensive movement. The second, or intermediate recovery phase, categorizes step type: 3 for dorsal stepping, 4 for plantar stepping. The final, or late recovery phase, gauges all other aspects of recovery: 5 for frequent plantar stepping with no coordination, 6 for frequent plantar stepping with some coordination, 7 for consistent plantar stepping with coordination and proper paw rotation, 8 for all the aforementioned criteria and mild trunk stability, 9 for full trunk stability and no tail dragging.

The gait acquisition and analysis system that has been developed is able to provide a BMS score over most of the scale, independent of the user. For many of the scores, it is additionally capable of providing a more accurate and qualitative assessment of the recovery criteria. Since the acquisition portion of the system was designed to detect plantar paw placement, scores lower than 4 are not delineated by the system. The early recovery scores which rank ankle movement are among the more subjective parts of the BMS. These scores depend on the operator to determine if joint movement is caused by voluntary muscle contraction or by the limb dragging across the surface. As this is difficult enough for a human operator to determine and not hugely significant to measuring recovery, only extent of injury, these scores were not automated in the analysis system. Additionally, score 3, dorsal stepping, is not included in the detection algorithm since the algorithm is focused on locating plantar steps for the collection of data relevant to computing gait metrics. Dorsal stepping could be determined if the proper target descriptor was added, but dorsal steps lack many of the characteristic paw features which the algorithm relies on to substantiate a match. The acquisition system does allow the user to review the recorded video and therefore these lower scores can be assigned manually if necessary.

The BMS scores 4-7 are more identifiable by the acquisition and analysis system as locating and documenting plantar steps and ascertaining interlimb coordination are the primary requirements of the system. Score 4, occasional plantar stepping, can be evaluated by examining a gait trace chart. Any traces appearing on the chart will indicate the paws that
made plantar placement. Scores of 5, 6, and 7 can be assigned based on the coordination number. Gait data with a coordination number of 0-6 and that show some plantar placement on the gait trace can be ranked as a 5. Data with a coordination number between 7 and 19 represent some coordinated stepping but may perhaps only show it with a single limb or with a slow response. This can be ranked as a 6. Data with high coordination numbers 20-30 show a considerable amount of interlimb coordination with a few slight stumbles or false starts. This also indicates consistent plantar stepping and proper paw rotation. Properly rotated paws increase the number of positively identified plantar steps since they are a closer match to the target descriptors. Data of this type can be ranked as a 7. Scores of 8 and 9 extend beyond what the analysis system is capable of ascertaining on its own. These scores rank trunk stability and tail dragging, two criteria which the detection algorithm does not look for.

Although only the middle four scores of the BMS are covered by the gait acquisition and analysis system, the system can still provide a better overall assessment of recovery. The goal of the research was not to create an automated BMS scoring device, but rather an advanced system capable of providing an objective method of evaluating animal recovery using quantifiable gait data. The BMS relies heavily on ranking interlimb coordination to measure recovery progress. This is a similarity both the BMS and the gait analysis system share. The gait analysis system, however, bases its coordination score on precise paw position and timing data rather than subjective observation. Additionally, the coordination number is weighted to emphasize quality of the coordinated stepping. Lastly, the coordination number is not grouped together with several other criteria to form an amorphous ranking. The weighted value and independence from other recovery criteria give greater significance to the value. A coordination number of 28 has very different meaning from a score of 45. A BMS score of 5 compared to a 6 is ambiguous when solely examining interlimb coordination.

The gait trace charts can be used as powerful tools to quickly estimate the level of injury or recovery. The charts provide a concise description of the measured timing data and strike order of the gait. BMS scoring for the middle ranges can be made directly from the chart information with no more subjective assessment than what's already inherent to the system. Additionally, the traces provide visual documentation of the gait that is easier to review and publish than a full video capture sequence. Comparing traces can also provide more agreement between observers on the behavior of the test animals than a score of a 5, 6, or 7 could.

The gait analysis system also reports spatial gait data, a set of measurements lacking from the BMS score. This data consists of paw placement relative to other paws, previous paw location, and the overall stride. It includes stride length, stance width, centerline variation, and all other measurements that can be computed based on precise paw location. These are all factors not considered by the BMS score but have been shown to be affected by injury. Acquired data shows that post-injury, the stride length decreases and the stance width increases. This is due to the short, quick steps and the increased weight support on the forelimbs. By simply comparing distance metrics before and after injury, a rough assessment of the animal's healthy can be made. These measures can indicate the overall strength and stamina of the mouse, and when combined with the rest of the gait data, determine the level of recovery with greater accuracy.

10.6 Refinement of the System

There are several aspects of the final system which could undergo further refinement to improve the quality and timeliness of the reported gait data. A better design of the walkway stand could allow for more strides to captured, thereby increasing the quality of the data and making the system less susceptible to variant mouse behavior. Improving the efficiency of the detection algorithm, primarily in the Hough transform can lead to decreased processing time and reporting of additional gait metrics. Solid-state force sensor technology could also be review again with the aim of creating a hybrid solution capable of collecting both high resolution positional data and force data.

10.6.1 Refinement of Camera/Plate Stand

There are several modifications to the stand which could be implemented. These include a longer walkway and a wide angle lens for greater strike coverage, better lighting to further reduce non-uniform intensities, a redesigned hood mount for better placement and light/dark selection, longer mouse houses to accommodate the animal's tail, and improved electrical wiring including a standardized connection block for the top half to be connected to the lower half or powered separately, a main power switch, and a switch to power the top level lights independently.

Limitations on the number of strides recorded per capture are due to the size and layout of the physical stand. A simple revision which could be made to accommodate the capture of more than three full strides (two usable) is the design a longer walkway. A longer walkway, however, requires a wider angle camera lens. The lens likely have to be custom ordered, thereby raising expensive and procurement time, since the focal length that would be required is not readily available for the size of mount and CCD of the camera. Additional image processing would also have to be done to correct for the increased image distortion. General modifications to the lighting, hood mount, and mouse house dimensions are among the user requested improvements. Improved lighting can be implemented to create background intensities with greater uniformity. This will serve to remove the dependence on digital gain and thereby decrease image noise. An better method of mounting the hood to the top of the device is most requested user modification. The current design requires precise alignment and does not remove and replace very quickly. This often leads to partial placement. It is also dependent on the user to place the correct side facing the plate for the color of mouse. A way to remove this requirement from the user would be desired in a revised design. Extending length mouse houses is needed to allow the mouse to fully enter, including its tail, into the enclosure. The current design did not take the tail into account and so there have been several instances where a tail gets pinched by the loading door or the walkway gate.

The device could also use improved wiring overall. Standardized connectors are needed to link the top and bottom portions of the device together. Currently, the connectors are flimsy and unprotected, allowing them to be easily bent. The connectors are also needed on the top walkway portion so that it may be powered independently during training. Additional circuitry is needed for the control of the top level lights as well. These lights are not used during black mice experiments and it would be preferable if the software could set them appropriately rather than the user. A main power switch is necessary too as the current method of powering the system is to simply plug it into the power outlet.

10.6.2 Refinement of Detection Algorithm

The detection algorithm can be modified to process the gait videos in a more timely manner and to detect and report paw rotation. It is possible that by implementing the full Hough transform, rather than the approximation, the computation time may decrease and rotation detection may be able to be added. The approximation method was originally implemented because it was more efficient for fewer target descriptors and provided higher quality results than the DoG method being used at the time. To ensure accurate matching though, the dynamic boundaries and multiple descriptors were implemented. This greatly increased the computation time. By reexamining the design and using the full Hough transform, it may be possible to streamline the process. The full Hough transform bases the descriptors on the local gradients which could help with avoiding false detections with white mice. This is because matching is done along the gradient edge rather than among a group of line segments which may randomly appear to be similar to a paw. Even if computation time is not diminished, it is theorized that rotation angle may be added without increasing run-time.

10.6.3 Alternative Designs

There are several alternative designs which may be explored during the continued development of this system. The two prominent physical designs are the single layer piezoelectric film sensor array and the camera/treadmill system. The single layer piezoelectric film sensor array would be based on the work done to measure force by measuring the change in natural resonance due to a strain. An array hundred of thousands tiny piezoelectric film elements would have the advantage of being able to detect both force and position. However, a considerable amount of work would have to be put into the design and fabrication. It also has the limitation of only being able to detect as many strides as can be covered in its length, just like the current system. The second alternative design, the camera/treadmill system, would function similar to the present design only the static plate would be substituted with a translucent, motor-driven, belt. A belt set at a constant velocity would ensure that the mouse is always moving consistently. This design also lessens the requirements on the camera. The camera now can focus on a much smaller area where lighting can be better controlled and the mouse has less area to wander in. By capturing only the length of a single stride, the frame rate could be tripled for the same resolution. Additionally, more strides would be captured per run leading to improved quality of gait measurements. Theoretically, this could provide enough data to create a general mathematical model of mouse gait.

10.7 Final Design Review

The final gait acquisition and analysis system utilizes a machine vision camera to record digital video of a mouse as it run across a translucent walkway. Once captured, the video is processed using an advanced, custom designed, image processing algorithm. The algorithm searches the video for paws in contact with the plate which appear as highly focused regions. The paws are identified by the algorithm as fore or hind, left or right and the center point and frame numbers are stored for each instance of contact. The data collected for each paw strike is reported to the user for further analysis. The final design meets the requirements of reliably detecting and identifying paws from a sequence of images, determining the location of each paw, associating instances of the same paw at the same location through time, and reporting the data to the user in a meaningful way. Additionally, the hardware is physically robust and portable. Mice are easily loaded and are completely contained within the device. The device is modular to facilitate cleaning and replacement of components.

There are some areas where the device falls short of meeting the full set of requirements. Most notably, the present design does not report rotation angle of the paws. Although the system is not incapable of determining paw rotation, this was decided to be omitted because of the dramatic increase in computation time required to calculate the data. The device is limited to capturing approximately three full strides of a healthy animal per run. This includes two full strides and a third being broken up across the beginning and ending of the sequence. The device is unable to measure force between the paws and the plate. Though this was not a requirement of the revised solution, it was an initial requirement of the device. This may still be implemented in the future. The computation time of the identification algorithm is extremely lengthy. All analysis is therefore done post-experiment.

Copyright © Casey Patrick Harr

Appendix I.

Basso Mouse Scale (BMS).

Early Recovery:

- 0 no ankle movement
- 1 slight ankle movement
- 2 extensive movement

Intermediate Recovery:

3 - plantar placement of the paw with or without wight support or dorsal stepping but no planter stepping

4 - occasional plantar stepping

Late Recovery:

5 - frequent or consistent plantar stepping, no coordination or frequent or consistent plantar stepping, some coordination, paws rotated at initial contact and lift off

6 - frequent or consistent plantar stepping, some coordination, paws parallel at initial contact or frequent or consistent plantar stepping, mostly coordinated, paws rotated at initial contact and lift off

7 - frequent or consistent plantar stepping, mostly coordinated, paws parallel at initial contact and rotated at lift off or frequent or consistent plantar stepping, mostly coordinated, paws parallel at initial contact and lift off and severe trunk instability

8 - frequent or consistent plantar stepping, mostly coordinated, paws parallel at initial contact and lift off and mild trunk stability or frequent or consistent plantar stepping, mostly coordinated, paws parallel at initial contact and lift off, normal trunk instability and tail down or up and down

9 - frequent or consistent plantar stepping, mostly coordinated, paws parallel at initial contact and lift off. normal trunk instability and tail always up

References

- [1] S.W. Scheff, A.G. Rabchevsky, I. Fugaccia, J.A. Main, and J.E. Lumpp Jr., "Experimental modeling of spinal cord injury: characterization of a force-defined inury device," *Journal of Neurotrama*, Vol. 20, 2003, pp.179-193.
- [2] J.E. Lumpp, K.N. Roberts, M. Wells, J.A. Main, C.P. Harr, O.A. Rawashdeh, and S.W. Scheff, "Characterization of a computer controlled non-penetrating cortical contusion device," *Journal of Neurotrauma*, Vol 20, 2003, pp. 1087.
- [3] K.N. Roberts, K.M. Miller, J.E. Lumpp, M. Wells, C.P. Harr, O.A. Rawashdeh, and S.W. Scheff, "Computer controlled cortical contusion device for the mouse," *Journal of Neurotrauma*, Vol. 21, 2004, pp. 1296.
- [4] D.M. Basso, M.S. Beattie, and J.C. Bresnahan, "A Sensitive and Reliable Locomotor Rating Scale for Open Field Testing in Rats," *Journal of Neurotrauma*, Vol. 12, No. 1, 1995, pp. 1-21.
- [5] D.M. Basso, L.C. Fisher, A.J. Anderson, L.B. Jakeman, D.M. McTigue, and P.G. POPOVICH, "The Basso mouse scale for locomotion (BMS) detects differences in recovery after spinal cord injury in five common mouse strains," *Journal of Neurotrauma*, in press.
- [6] G.C. Koopmans, R. Deumens, W.M.M. Honig, F.P.T. Hamers, H.W.M. Steinbusch, and E.A.J. Joosten, "The assessment of locomotor function in spinal cord injury rats: the importance of objective analysis of coordination," *Journal of Neurotrauma*, Vol 22, No. 2, 2005, pp. 214-225.
- [7] S.W. Scheff, D.A. Saucier, M.E. Cain, "A statistical method for analyzing rating scale data: the BBB locomotor score," *Journal of Neurotrauma*, Vol. 19, No. 10, 2002, pp. 1251-1260.
- [8] K.A. Clarke, and J. Still, "Gait analysis in the mouse," *Physiology & Behavior*, Vol. 66, No. 5, 1999, pp. 723-729.
- [9] K.A. Clarke, and J. Still, "Development and consistency of gait in the mouse," Vol. 73, 2001, pp.159-164.
- [10] T.G. Hampton, M.R. Stasko, A. Kale, I. Amende, A.C.S. Costa, "Gait dynamics in trisomic mice: quantitative neurological traits of Down syndrome," *Physiology* & *Behavior*, Vol. 82, 2004, pp. 381-389.

- [11] 3M, Resistive Design and Integration Guide. http://www.3m.com/3mtouchsystems/Products/Resistive/index.jhtml
- [12] Interlink Electronics, FSR: Force Sensing Resistor Integration Guide and Evaluation Parts Catalog, Ver 1.0, Rev D. <u>http://www.interlinkelec.com</u>
- [13] Pressure Profile Systems, *TactArray*. http://www.pressureprofile.com/tactArray.php
- [14] Synaptics, *Capacitive Position Sensing*. http://www.synaptics.com/technology/cps.cfm
- [15] Elo Touchsystems, *Surface Wave Acoustic Touchscreens*. http://www.elotouch.com/products/inteltec/intelwork.asp
- [16] Tactex Controls, *Kinotex Touch Textiles*. <u>http://www.tactex.com/technology.htm</u>
- [17] Measurement Specialties, *Piezo Film Sensors Technical Manual*, Rev B, April 1999. <u>http://www.msi.com</u>
- [18] Measurement Specialties, LDTO Solid State Switch/Vibration Sensor, Application Note 111398, Rev 0. <u>http://www.msi.com</u>
- [19] G. Murali Krishna and K. Rajanna, "Tactile sensor based on piezoelectric resonance," *IEEE Sensors Journal*, Vol. 4, No. 5, October 2004, pp. 691-697.
- [20] T.V. Papakostas, J. Lima, M. Lowe, "A large area force sensor for smart skin applications," *Conference Proceedings*, 1st IEEE Sensors Conference, Orlando, FL, June 2002.
- [21] Mouse Specifics, Inc., "Gait Analysis System for Rodents," *DigiGait Product Sheet*. <u>http://www.mousespecifics.com/DigiGait Product Sheet.pdf</u>
- [22] PixeLINK, *PL-A741 Machine Vision Camera System Guide*, Ver. 4.0, April 2004. http://66.148.40.196/uploads/PL-A741_SYSTEM_GUIDE.pdf
- [23] PixeLINK, Application Programming Interface Reference Manual, Version 4.1, Doc No. 04326-02, July 2004. <u>http://66.148.40.196/uploads/PL-A741_USERS_MANUAL.pdf</u>

- [24] Edmund Industrial Optics, "Terminology used in electronic imaging," *Optics and Optical Instruments Catalog*, Vol. N048A, 2004, pp. 204, 229.
- [25] Silicon Laboratories, *C8050F300/1/2/3/4/5 Reference Guide*, Rev. 2.6, 2005.
- [26] D.G. Lowe, "Object recognition from local scale-invariant features," *International Conference on Computer Vision*, Corfu, Greece, September 1999, pp. 1150-1157.
- [27] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, Vol. 60, No. 2, 2004, pp. 91-110.
- [28] E.L. Walker, "Binary Image Processing," CPSC 320: Computer Vision Class Notes, Week 2, Fall 2004. <u>http://cs.hiram.edu/~walkerel/cs320/lectures/binary.ppt</u>
- [29] B.S. Morse, "Segmentation: The Hough Transform," 1995. http://visl.technion.ac.il/labs/anat/12-Hough/
- [30] L.C. Mai, "The Hough Transform," Introduction to Computer Vision and Image Processing, Ch. 6, Sec. 2. <u>http://www.netnam.vn/unescocourse/computervision/62.htm#6.2 Hough</u> <u>Transfrom</u>
- [31] D.H. Ballard, "Generalizing the Hough transform to detect arbitrary patterns," *Pattern Recognition*, Vol. 13, No. 2, 1981, pp. 111-122.

Vita

Data of Birth: November 15, 2005 Place of Birth: Gallipolis, Ohio

Education:	Bachelor of Science in Electrical Engineering (BSEE).
	University of Kentucky, May 2003.
	Lexington, Kentucky
Positions:	Research and Development, 2002 - 2005,
	Precision Systems and Instrumentation, LLC.
	Research Assistant, Spring 2003 - Spring 2005,
	Department of Electrical and Computer Engineering,
	University of Kentucky.
	Software Developer, Summer 2003,
	Two Dimensional Instrumentation, LLC.
	Electronics Department Supervisor, Summer - Fall 2001,
	Office Max.

Publications:

J.E. Lumpp, K.N. Roberts, M. Wells, J.A. Main, C.P. Harr, O.A. Rawashdeh, and S.W. Scheff, "Characterization of a computer controlled non-penetrating cortical contusion device," *Journal of Neurotrauma*, Vol 20, 2003, pp. 1087.

K.N. Roberts, K.M. Miller, J.E. Lumpp, M. Wells, C.P. Harr, O.A. Rawashdeh, and S.W. Scheff, "Computer controlled cortical contusion device for the mouse," *Journal of Neurotrauma*, Vol. 21, 2004, pp. 1296.

Casey Harr