(ABSTRACT)

Title of Document: DEVELOPMENT OF A QUADRUPED ROBOT AND PARAMETERIZED STAIR-CLIMBING BEHAVIOR

Thomas Kevin Brewer, M.S., 2011

Directed By: Professor Satyandra K. Gupta
Department of Mechanical Engineering
Institute for Systems Research

Stair-climbing is a difficult task for mobile robots to accomplish, particularly for legged robots. While quadruped robots have previously demonstrated the ability to climb stairs, none have so far been capable of climbing stairs of variable height while carrying all required sensors, controllers, and power sources on-board. The goal of this thesis was the development of a self-contained quadruped robot capable of detecting, classifying, and climbing stairs of any height within a specified range. The design process for this robot is described, including the development of the joint, leg, and body configuration, the design and selection of components, and both dynamic and finite element analyses performed to verify the design. A parameterized stair-climbing gait is then developed, which is adaptable to any stair height of known width and height. This behavior is then implemented on the previously discussed quadruped robot, which then demonstrates the capability to climb three different stair variations with no configuration change.

DEVELOPMENT OF A QUADRUPED ROBOT AND PARAMETERIZED STAIR-CLIMBING BEHAVIOR

By

Thomas Kevin Brewer

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Masters of Science
2011

Advisory Committee:
Professor Satyandra K. Gupta, Chair/Advisor
Professor Hugh A. Bruck
Assistant Professor Sarah Bergbreiter

© Copyright by
Thomas Kevin Brewer
2011

Acknowledgements

First, I would like to thank Dr. Gupta for all the time and effort he has put into helping me not only with this project, but with everything I've done at University of Maryland. Without his valuable input and advice, I would never have completed this effort successfully or on time.

I would also like to thank Dr. Krishnanand Narasimha for all of his assistance on this project, particularly with shaping the direction and presentation of this work, as well as a large amount of assistance on the vision portion of the project. I appreciate all the time he has taken out of his busy schedule to work on this project, and I look forward to working with him more in the future.

I also owe a large debt of gratitude to Andrew Vogel, for all of his assistance on the programming and control aspects of this work. When I started on this project in January, I was very controls and programming illiterate, and I have relied heavily on Andrew to assist me when I could track down illusive programming bugs.

All of the other lab-mates in the Advanced Manufacturing Laboratory have also provided a large amount of help and moral support, particularly Bill, Beth, David, and Greg. Their assistance on my project and with schoolwork in general has been invaluable.

I would also like to thank the United States Navy for providing me with an amazing career and the opportunity to obtain my Masters.

Finally, I would like to thank my parents, for the fantastic advice and unconditional love they have always given me, and Julie, for toughing it out with me even through the long hours and weekends spent at the lab.

Table of Contents

Acknowledgements ii
List of Tables........ v
List of Figures vi
Chapter 1 - Introduction........ 1
1.1 Background........ 1
1.1.1 Motivation for Stair Climbing Robots 1
1.1.2 Advantages/Disadvantages of Quadruped Robots........ 3
1.2 Motivation 4
1.2.1 Mechanical Design........ 4
1.2.2 Parameterized Behavior Design........ 5
1.2.3 Sensors and Controllers........ 5
1.3 Thesis Goal and Scope........ 6
1.4 Organization 7
Chapter 2 - Related Work 9
2.1 Overview 9
2.2 Stair Climbing Mechanisms 10
2.2.1 Biped Robots........ 10
2.2.2 Tracked Robots 11
2.2.3 “Hybrid”, Wheeled, and Hexapod Robots........ 12
2.3 Quadruped Stair Climbing Robots........ 13
2.3.1 TITAN VI 14
2.3.2 ROACH 17
2.3.3 SCOUT-1/SCOUT II........ 19
2.3.4 HIT-HYBTOR........ 22
2.3.5 Quattroped 25
2.3.7 LittleDog........ 27
2.4 Vision and Sensor Based Stair Detection........ 31
2.4.1 Stereo Vision Based Detection 31
2.4.2 Range-Finder Based Detection 33
2.4.3 Image Processing Based Detection 35
2.4.4 Microsoft Kinect in Robotics........ 37
Chapter 3 – Design of Quadruped Robot........ 39
3.1 Overview 39
3.2 Design Requirements and Specifications 39
3.3 Design Concept........ 41
3.3.1 Joint Configuration........ 42
3.3.2 Robot Configuration........ 45
3.4 Selection of Commercial-Off-the-Shelf Components 46
3.4.1 Actuators........ 46
3.4.2 Sensors........ 51
3.4.3 Batteries........ 55
3.4.4 Controllers 56
3.5 Design of Custom Components........ 58
3.6 Simulation-Based Sizing of Components 64

3.6.1 Dynamics Modeling 64
3.6.2 Finite Element-Based Modeling 69
3.7 Summary 74
Chapter 4 – Design and Testing of Parameterized Stair-Climbing Behavior 79
4.1 Overview 79
4.2 Kinematic Analysis 81
4.2.1 Leg Analysis 81
4.2.2 Full body Analysis 87
4.3 Behavior Concept 90
4.3.1 Overall Behavior Design 90
4.3.2 Phase One 90
4.3.3 Phase Two 93
4.3.4 Phase Three 95
4.4 Behavior Parameterization 97
4.4.1 Stair Height Parameterization 97
4.4.2 Stair Width Parameterization 99
4.5 Behavior Execution 101
4.5.1 Stair Parameter Estimation 101
4.5.2 Executing Behavior 104
4.6 Experimental Results 106
4.6.1 Experimental Setup 106
4.6.2 Results 107
4.7 Summary 111
Chapter 5 – Conclusions 113
5.1 Intellectual Contributions 113
5.2 Anticipated Benefits 114
5.3 Future Work 115
5.3.1 Improved Actuator Performance 115
5.3.2 Kinect Data Processing Improvement 115
5.3.3 Behavior Development and Parameterization for Small Stairs 116
Bibliography 117

List of Tables

Table 1 HiTec HS-805BB Analog Servo Specifications [63] 49
Table 2 Microsoft Kinect Sensor Specifications [64] 51
Table 3 Arduino Mega 2560 Specifications [69] 56
Table 4 13-in Macbook Air Specifications [70] 57
Table 5 Delrin Acetal Resin Material Properties [71] 58
Table 6 Leg DH Parameters 84

List of Figures

Figure 1 Honda Asimo [7] and HRP-2 [5] Climbing Stairs 11
Figure 2 Tracked Stair-Climbing [8]. 12
Figure 3 (a) Loper [12], (b) ASGUARD [13] and (c) MSRox [14]..........................12
Figure 4 PV-II Climbing Stairs [25]..14
Figure 5 TITAN VI Leg Mechanism [21] 15
Figure 6 TITAN VI Robot Climbing Stairs [26]..16
Figure 7 ROACH Leg Mechanism [27] 18
Figure 8 ROACH Quadruped Robot [29]..19
Figure 9 SCOUT Climbing Stairs [35]..21
Figure 10 SCOUT-II [36] 22
Figure 11 HIT-HYBTOR Leg Configuration [38]..23
Figure 12 HIT-HYBTOR Climbing Stairs [38] 24
Figure 13 Quattroped with Hybrid Legs [39] 26
Figure 14 Quattroped Climbing Stairs with Hybrid Legs [20] 27
Figure 15 Cross-section View of LittleDog [51] 28
Figure 16 LittleDog Stair-Climbing Utilizing a Dynamic Gait [45].........................30
Figure 17 LittleDog Stair-Climbing Utilizing a Static Gait [52] 30
Figure 18 QRIO Robot [54]...32
Figure 19 Stereo Vision Recorded by BARt-UH [55] 33
Figure 20 Results of ASTERISK's Range-Finder Scan [15] 34
Figure 21 TMR Stair Detection [56] 36
Figure 22 Robot Leg Joint Configuration...43
Figure 23 Leg and Body Configuration..45
Figure 24 Servo Motor Cut-away View [61] 47
Figure 25 (a) HiTec HS-805BB Servo (b) Dimensioned Drawing of HS-805BB [63] 50
Figure 26 Microsoft Kinect Sensor [64]...52
Figure 27 Voltage Regulator Schematic [66] 53
Figure 28 Completed Voltage Regulator and Kinect Power Cable...........................53
Figure 29 HiTec H7 Infrared Distance Sensor [67] 54
Figure 30 1.2 Volt C-Cell Tenergy Battery [68]...55
Figure 31 Arduino Mega 2560 Microcontroller [69] 56
Figure 32 Delrin Acetal Resin in Sheet Form [72] 58
Figure 33 ProE Linkage Model...60
Figure 34 Joint-Assist Springs 61
Figure 35 Leg-Assist Springs Fully Extended 62
Figure 36 Compliant Lower Leg and Foot Mechanism..63
Figure 37 Dynamic Model With Constraints...65
Figure 38 Initial Position for Simulation 66
Figure 39 ProE Simulation Results: (a) Time=0 sec, (b) Time=0.1 sec, (c) Time=0.2 sec 68
Figure 40 Upper Component of Spring-Leg-Foot Mechanism.................................69
Figure 41 Simulation Constraints..71
Figure 42 Static Load FEA Results 73

Figure 43 Maximum Stress Location 74
Figure 44 Lift Joint Linkage 75
Figure 45 Front-Right Leg Assembly..76
Figure 46 Assembled Robot, Front and Back 77
Figure 47 Leg Reference Coordinate Systems..82
Figure 48 Body Dynamics 87
Figure 49 Robot Initial Position 91
Figure 50 Body Lift Position 92
Figure 51 Front Legs Up Position 93
Figure 52 Body Forward Position 94
Figure 53 Legs Forward Position 95
Figure 54 Rear Legs Up Position 96
Figure 55 Standing Position...97
Figure 56 Parameterized Left Rear Lift Joint Angles 99
Figure 57 Kinect Depth Sensor Image...102
Figure 58 Stair Edge Detection Image...103
Figure 59 Robot Successfully Climbing 2.5 Inch Stair 108
Figure 60 Robot Successfully Climbing 4 Inch Stair 109
Figure 61 Robot Successfully Climbing 5 Inch Stair 111

Chapter 1 - Introduction

1.1 Background

1.1.1 Motivation for Stair Climbing Robots

Robots have an increasingly pervasive presence in today's world, from large manufacturing applications utilizing specialized arm-type robots all the way to small toy robots for children. With advancements in computing power and miniaturization of computers and consumer electronics, the possible level of complexity of robots has increased as well. In fact, the question in normally asked in modern robotics has changed from "Can a particular task be accomplished by a robot?" to "How well can a particular task be performed by the most efficient robot?" This has led to the development of unique, multi-purpose robots capable of undertaking a large variety of complex tasks, utilizing advanced control systems as well as some levels of autonomy.

While this trend has applied to most areas of robotics, it has been most prevalent in the Military and Search and Rescue (SAR) applications. These two fields lend themselves particularly well to both Autonomous and Mobile robots.

Mobile robots are defined as robots which can, under their own power, move themselves in their environment [1]. They are usually designed for specific environments, such as deserts, urban settings, or even underwater. Mobile robots have been designed to travel using a large variety of locomotion methods, including any number of legs, wheels, tracks, and even snake inspired robots [2]. All of these methods have both advantages and disadvantages directly related to the type of terrain they are designed to traverse.

While many challenging tasks have been performed by mobile robots of many different designs, one task which stands out as a defining test of a robot's capability to traverse difficult terrain is the ability to climb stairs. While not a naturally occurring terrain feature, stairs and steps are widely placed throughout the world, particularly in urban environments, and therefore a mobile robot's ability to climb stairs can be crucial to expanding its available operating area.

Therefore, a useful test for determining the mobility of a robot is to determine if it is capable of stair climbing, and if so what is the size and number of stairs which can be climbed.

For this thesis, it is desired to develop, manufacture, and test a highly capable stair-climbing robot of the smallest possible overall volume which is capable of climbing stairs of several different sizes, while also able to determine the information about the stairs necessary to climb them utilizing only sensors and controllers carried onboard the robot. As will be discussed further in Chapter 2, many different types of mobile robots are capable of stair-climbing, however, as will be shown in Chapter 2, a self-contained stair-climbing quadruped robot has yet to be developed. Therefore, a quadruped design will be developed and utilized in this work.

1.1.2 Advantages/Disadvantages of Quadruped Robots

Quadruped robots can be a highly mobile and robust platform, but like all mobile robot platforms, they have both advantages and disadvantages. One of the main advantages of quadruped robots is their ability to negotiate terrain other platforms are unable to traverse. For example, while tracked or wheeled mobile robots generally have much higher locomotion speeds, legged platforms are capable of precise foot placement which would allow them to traverse gaps or barriers that wheeled or tracked platforms are unable to negotiate.

While improved terrain negotiation is an advantage of all legged platforms over wheeled and tracked platforms, quadrupeds have several particular advantages over other legged platforms with different numbers of legs. A quadruped robot is capable of utilizing a static gait, in that it is statically stable while moving, by always having at least three feet in contact with the ground at a given time. This is inherently more stable that a bipedal robot, which in order to move must have only one foot in contact with the ground at any one time. While quadrupeds are generally more stable than bipedal robots, they have advantages when compared to robots with more than four legs as well. While increasing the number of legs of a mobile robot increases the robot's stability, it also increases the complexity of the control and gait required for motion. The increase in the number of legs also increases the weight of the robot, which decreases the payload capacity and efficiency of the robot. A quadruped robot has the minimum number of legs necessary to utilize a static gait, and for this application this means that the efficiency is maximized.

The strengths of a quadruped design are closely related to some of its biggest weaknesses. While less complicated to control than robots with more than four legs, a quadruped robot is still much more complex to control than a tracked or a wheeled robot, requiring more controlled degrees of freedom to achieve the same locomotion that a wheeled or tracked platform can accomplish with half that number. Along the same lines, a legged robot requires much more power to achieve the same speed of locomotion that a wheeled or tracked platform, if comparable speeds are even achievable, which in most cases they are not. Legged platforms may also require power just to maintain the position of their body when at rest, since depending on the robot's configuration some joints must constantly be powered to maintain any controlled position.

Despite these disadvantages, legged platforms in general have the potential to outperform wheeled or tracked platforms in many applications, including stair-climbing, and therefore it is important to continue to investigate legged platforms, despite their current limitations in some areas.

1.2 Motivation

The motivation for this thesis can be categorized into self-contained stair-climbing quadruped design, parameterized quadruped stair-climbing behavior design, and stair detection sensor and robot control design.

1.2.1 Mechanical Design

As will be discussed in Chapter 2, there are many quadruped robot designs which are capable of stair-climbing, however no current or past stair-climbing quadruped design contains all components required for stair-climbing, with some combination of the robot power source, controller, or sensors being located external to the robot itself. While stair-climbing capability is not a definitive test for mobile robots, a robot which is capable of stair-climbing is generally capable of traversing rugged terrain as well, therefore the development and construction of a quadruped robot which is capable of stair-climbing with all necessary components located on the robot itself will be discussed.

1.2.2 Parameterized Behavior Design

There are many approaches to the problem of stair-climbing for legged robots, and most require specialized mechanisms which are stair-climbing specific and do not contribute the to the overall performance of the robot or the development of complicated and computationally intensive gait planning and execution. To avoid the disadvantages of both of these approaches, a static quadruped stair-climbing behavior will be developed, which will then be parameterized to allow the quadruped to climb stairs of various heights and widths.

1.2.3 Sensors and Controllers

While focus of this work is the development of a self contained quadruped robot capable of climbing stairs, it is also desired to develop a platform which is capable of more tasks than just stair-climbing. Therefore the sensor and control system used to detect and classify the stairs must also be capable of more than just stair detection. As shown in Chapter 2, several methods of stair detection have been implemented in previous stair-climbing robots, however these methods are either so specialized that they are not capable of detecting any other environmental objects, or they are too large and heavy to be implemented on a self-contained robot. For this reason, a new lightweight sensor and control system will be developed which, while capable of stair detection, will also be capable of detecting the environment in general, which will allow the robot developed to be utilized for tasks other than stair climbing.

1.3 Thesis Goal and Scope

The goal and scope of this thesis can be organized into the topics laid out in Sections 1.2.1 through 1.2.3. First, a leg configuration which will allow for both a high load capacity as well as a large operating space will be developed. These legs will then be combined with a body designed to accommodate stair climbing to create the overall joint configuration of a stair-climbing quadruped robot. To construct a robot of this configuration, commercially available actuators, sensors, batteries, and controllers will then be selected. These components will then be used together with custom-made components to fully describe the robot. Next, to ensure proper component selection and robot component sizing, a dynamic model of the robot will be constructed to ensure the robot is capable of stair-climbing. To examine the structural reliability of the robot a Finite Element Analysis will then be performed on the component of the robot identified to be most at risk of failure. Once this analysis is complete, the robot will be constructed.

Once the physical robot is complete, a full kinematic analysis of the leg mechanism will be perform, which will then be applied to develop a kinematic analysis of the robot body. Next a static stair-climbing behavior will be developed, which will allow the constructed robot to climb stairs with a height of 5 inches and a width of 10 inches. This behavior will then be parameterized to allow the robot to climb stairs of between 2.5 and 5 inches in height, with widths of 10 to 12 inches. To utilize this parameterized behavior, a sensor and controller system will be developed for the robot which will allow it to detect both the height of stairs between 2.5 and 5 inches in height, as well as the width of each step. This system will then be used to implement the parameterized behavior. The robot, parameterized behavior, and sensor system will then be tested and the results reported. The complete robot should meet the following goals:

1) The robot will be capable of carrying all sensors, controllers, and batteries necessary to detect, classify, and climb stairs.
2) The robot will be capable of utilizing a parameterized behavior to climb any stair with height between 2.5 and 5 inches and width between 10 and 12 inches.
3) Utilizing self-contained sensors, the robot must be capable of accurately measuring the height and width of stairs.

1.4 Organization

Chapter 2 is a survey of the literature pertaining to stair-climbing robots. Bipedal, tracked, wheeled, Hybrid, and hexapod type robots are first discussed. This is followed by a thorough discuss of seven of the most relevant quadruped stair-climbing robots which have been previously developed. These are examined in depth, and show that none are capable of self-contained stair-climbing, a survey of sensors and methods used for stair detection and classification, followed by a survey of the usage of the Microsoft Kinect sensor and its published use in mobile robotics.

Chapter 3 contains a complete description of the robot design process. The joint configuration and body configuration are both developed and discussed. The selection of all robot components is then discussed, including the selection of the actuators used to drive the robot's joints, the sensors used to detect stairs, batteries to power both the servos and the sensors, and the controllers used for both sensors data processing and robot behavior control. The design of custom components, including the main construction materials and unique joint design concepts, are then discussed. The dynamic and finite element analysis performed on the robot is then delineated, followed by a summary of the robot construction, including the final construction results.

Chapter 4 discusses the development of a parameterized stair-climbing behavior. To fully describe the robot, a kinematic analysis of one of the robot's legs, as well the analysis of the robot as a whole is described. To overall behavior concept is then developed, with a detailed description of the static stair-climbing behavior included. The process by which this behavior is parameterized to be utilized for any stair height is then discussed, followed by the parameterization process to account for stair width. The process by which this behavior is implemented for the quadruped robot described in Chapter 2 is then presented. This includes a detailed description of the implementation of the Kinect sensor as the means of stair detection and classification. Finally, the robot's stair-climbing abilities are determined utilizing a described experimental program, and the results are reported.

Chapter 2 - Related Work

2.1 Overview

To best provide the background and motivation for the goal of this thesis, it is appropriate to investigate the various methods which have been used to develop and implement mobile robots capable of, or designed to, climb stairs. First, bipedal, tracked, wheeled, hexapod, and other various “hybrid” locomotion robots will be discussed. While the focus of this thesis is quadruped locomotion, principles derived from other platform’s efforts at stair-climbing are useful and related to quadruped stair-climbing design. Second, successful quadruped stair-climbing robots will be discussed, including both some of the first efforts in the field as well as some of the most recent. This is vital to understand both the scope of the research involved as well as the basis of some of the principles used in this effort. The third section addresses the different methods used to recognize and derive the parameters of stairs. While the main focus of this thesis is the development of the quadruped robot itself and the behavior by which it ascends stairs, the methods through which each stair-climbing platform attempts to perform this task autonomously though the use of sensors are also related to the overall goal and will be surveyed. Finally, previous work with the Kinect sensor will be discussed to provide both the background and motivation for its use.

2.2 Stair Climbing Mechanisms

2.2.1 Biped Robots

Bipedal robots, which walk utilizing two single to multiple degree of freedom legs, and are generally designed to imitate human locomotion. Bipedal robot locomotion has always been challenging, both due to the balance issues involved with only have two points of contact with the ground, as well as the inherently dynamic nature of bipedal locomotion. While humans generally walk utilizing a dynamic gait, most bipedal robots to date use statically stable gaits. This is achieved by having a comparatively large foot area and shifting the center of gravity of the robot over a statically placed foot prior to the movement of the other foot.

This strategy has generally been the type implemented for bipedal robotic stair climbing. Researchers at the University of Cassino, Italy have developed several bipedal robots capable of both ascending and descending stairs [3], [4]. Their EP-WAR2 and EP-WAR3 robots utilize a Programmable Logic Controller to climb stairs using a static gait. Another bipedal which utilizes a static gait for stair climbing is the HRP-2, used by researchers at Carnegie Mellon University and the National Institute of Advanced Industrial Science and Technology in Tokyo Japan [5]. This robot, while highly advanced and multi-task capable, also utilizes a static stair-climbing gait. Even Honda's well-known Asimo utilizes a static gait as well [6]. Asimo (left) and HRP-2 (right) are shown below utilizing their stair-climbing capabilities in Figure 1.

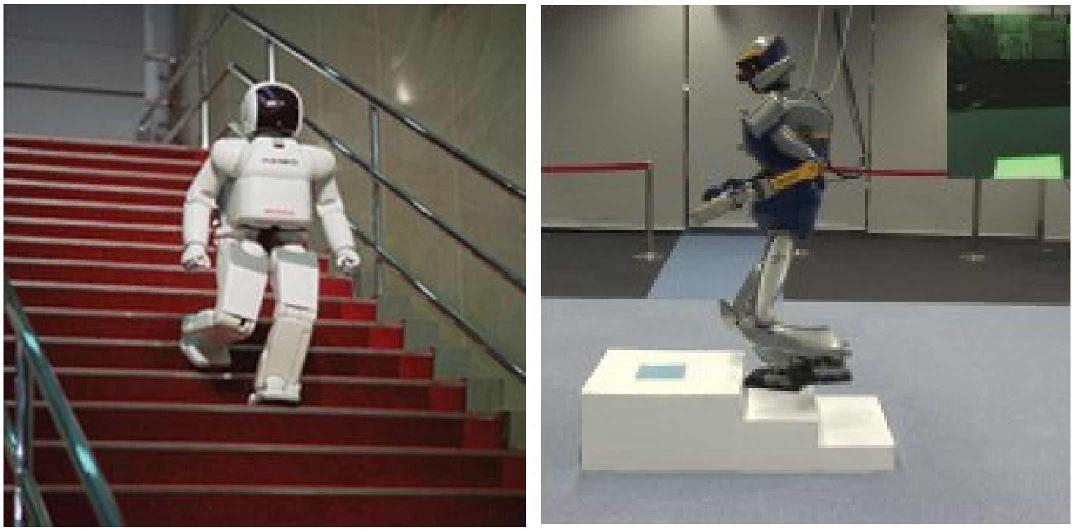


Figure 1 Honda Asimo [7] and HRP-2 [5] Climbing Stairs

2.2.2 Tracked Robots

Another platform for which stair-climbing has been thoroughly investigated is the tracked mobile robot. Typically, these robots have a long wheel-base and high torque and traction which allows them to climb stairs by driving up the stairs, with contact between only the treads and the edges of the stairs. However, most small mobile tracked robots require some type of extra actuation to climb stairs, or at least to position the robot on the first step prior to climbing. A typical example of this is the robot developed by researchers at the University of Minnesota, shown in Figure 2 [8].

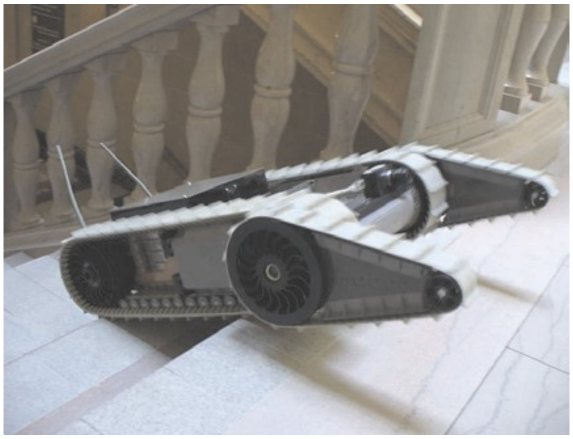
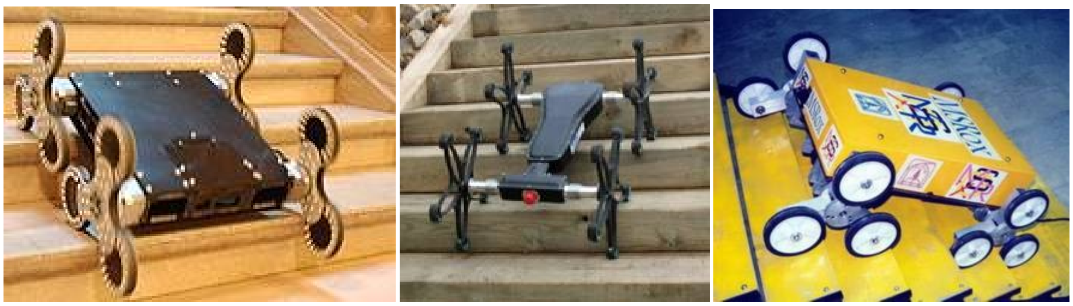


Figure 2 Tracked Stair-Climbing [8].

These types of robots are highly versatile, and have been demonstrated to both climb stairs while driven by an operator as well as autonomously.

2.2.3 "Hybrid", Wheeled, and Hexapod Robots

While there are many control solution to enable robots to climb stairs using legs, another possible solution is to equip the robot with specialized locomotive apparatus specifically designed to climb stairs. Some, like Loper [9], ASGUARD [10], and MSRox [11] use lobed wheel designs which allow them to utilize the geometry of their wheels to ascend stairs, as well as traverse rugged terrain, where normal round wheels would not achieve the same results.



(a) (b) (c)

Figure 3 (a) Loper [12], (b) ASGUARD [13] and (c) MSRox [14]

One very common goal in stair climbing is to construct a wheelchair or seat on which a person can sit and be transported up a staircase. These are generally very specialized mechanisms, and not adaptable for other uses, so an in-depth study of the related literature in this subject will not be presented.

Finally, extensive work has been documented on stair-climbing hexapods. A traditional hexapod with multiple degrees of freedom per leg, ASTRISK [15] has 24 degrees of freedom and will be discuss later in this chapter. One of the more well-known hexapod stair-climbing robots is RHex [16], which has 6 compliant legs with one degree of freedom each. The RHex is capable of traversing very rough terrain as well as stairs by simply varying the rate and sequence with which it rotates its legs. A very similar robot, the Quattroped [17–20], will be discussed later in this chapter.

2.3 Quadruped Stair Climbing Robots

As previously mentioned, stair-climbing is a commonly attempted task for mobile robots, and quadrupeds are no exception. Stair climbing with quadruped robots has been attempted since the beginning of robotic design, with some of the first successes being accomplished by researchers at the Tokyo Institute of Technology with PV-II robot shown in Figure 4, as well as the TITAN series of robots[21–24], of which TITAN-VI will be discussed in more detail later in this chapter.

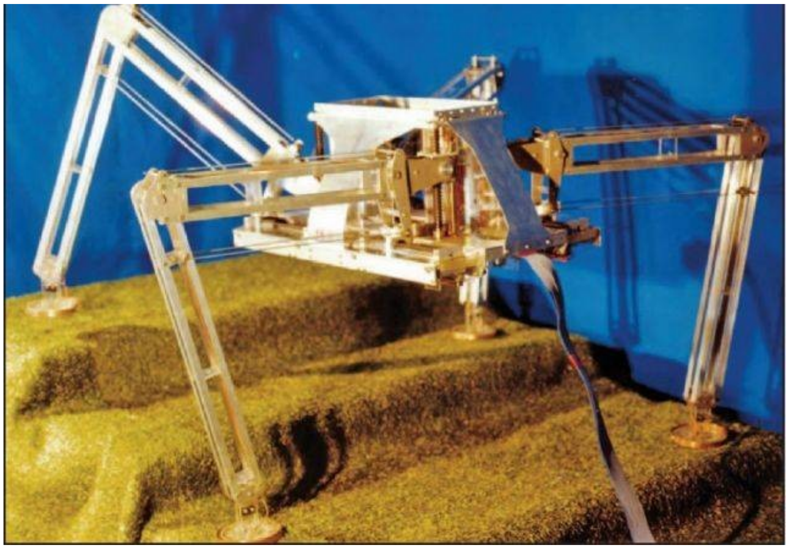


Figure 4 PV-II Climbing Stairs [25]

While many quadrupeds have been successful in stair-climbing, as will be demonstrated throughout this section, none to date have been successful in climbing stairs of any size (within kinematic capabilities) without the use of external computing/controllers, external sensors, external power source, an operator's input to "drive" the robot up the stairs, or a combination of the four. What follows is a survey of a selection of quadruped robots which have successfully exhibited stair-climbing and a brief description of their designs and control mechanisms.

2.3.1 TITAN VI

From 1990 to 1994, Hirose et al at the Tokyo Institute of Technology developed the quadruped robot TITAN VI, part of their series of TITAN robots [21]. Their goal was to create a robot which could climb stairs normal sized stairs of between 30 and 40 degrees of incline, as well as walk using both a static and dynamic gait. The TITAN VI utilizes a unique leg mechanism, shown in Figure 5, to achieve four degrees of freedom per leg. The leg design consists of an Evans mechanism and a slider-crank mechanism driven by ball screws, which achieve motion in the horizontal direction away from the body as well as rotation in the horizontal plain, in addition to vertical motion. A telescopic leg mechanism contained within the leg itself provides additional motion in the vertical direction, and the last degree of freedom is the passively actuated foot, which is kept with its bottom surface in the horizontal plain by a system of pulleys and a tension spring. With this passive attitude control, it is assured that the foot maintains maximum contact with the ground at all times.

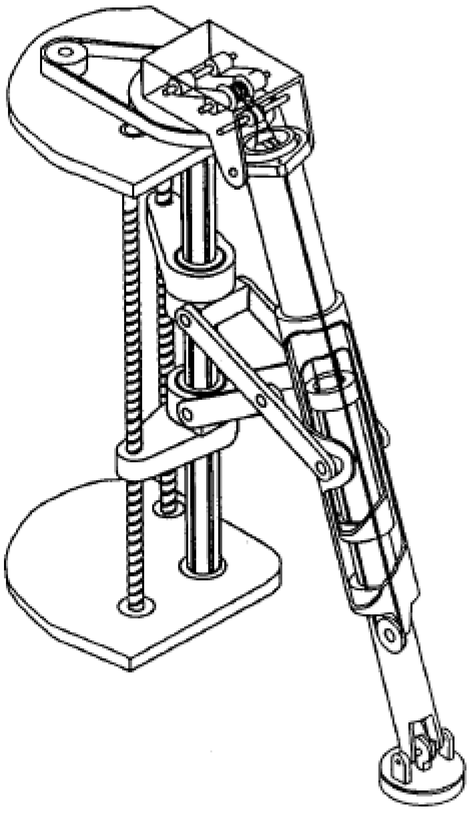


Figure 5 TITAN VI Leg Mechanism [21]

The leg also utilizes a dual mode transmission in the actuation of the vertical motion which allows for high speed motion while not under load, but contains an electromagnetic clutch to support the weight of the robot when loaded. This leg configuration makes extensive use of mechanical advantage to maximize the weight capacity and energy conservation of the robot. Another feature of this robot shown in Figure 6 is the articulated sliding joint placed in the center of the main body of the robot, which allows the body to be split to maintain the horizontal attitude of both body sections.

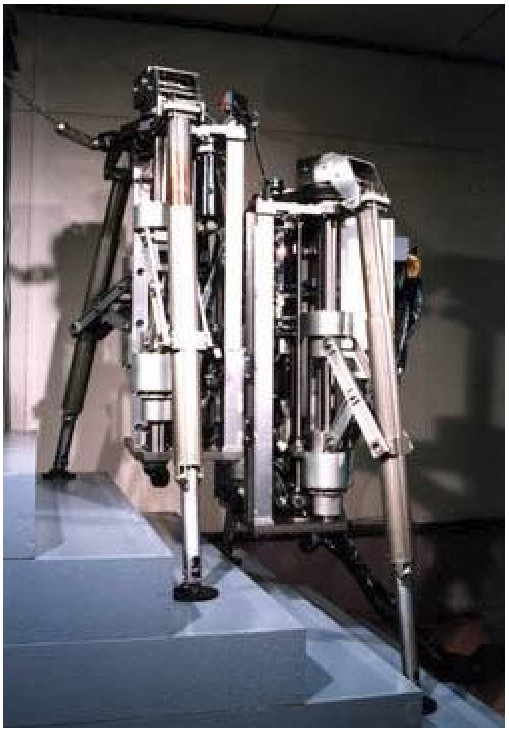


Figure 6 TITAN VI Robot Climbing Stairs [26]

The TITAN's actuators are driven by electric motors, and its power source is connected by umbilical, as is its control computer. It utilizes a static gait control to climb stairs, and while tunable to climb stairs of various sizes, it has no onboard capability to tune its gait to conform to different stair heights.

While the TITAN VI is highly capable of climbing stairs of heights actually found in architecture, it also has several limitations. Its large size (700(L) x 550(W) x 1060(H) mm) limits the width of the stairs that it can accommodate, as well as hallways and tight spaces. It also has a very large weight of 195 kg, and power requirements of 120 watts DC for each leg.

The TITAN VI introduced revolutionary concepts for stair-climbing robots, but due to the limitations of the technology of its time, it has been surpassed in many areas by more modern robots.

2.3.2 ROACH

The ROACH robot was developed from 1994 to 1996 by Daniel Pack at the Purdue Robot Vision Lab as a quadruped robot capable of static and dynamic gaits as well as vision-based stair climbing [27], [28]. Their main area of research was the interaction and coordination between high-level and low-level controllers, as well as how to integrate sensors into gait control.

ROACH has 12 degree of freedom, with three joints per leg. Each leg has two joints with rotation in the horizontal plain connected by a linkage, with a third linear joint attached to the end of the leg which actuates foot motion in the vertical direction. The full leg configuration allows for full control of leg positioning in the horizontal plain as constrained by the kinematics, and is shown in Figure 7.

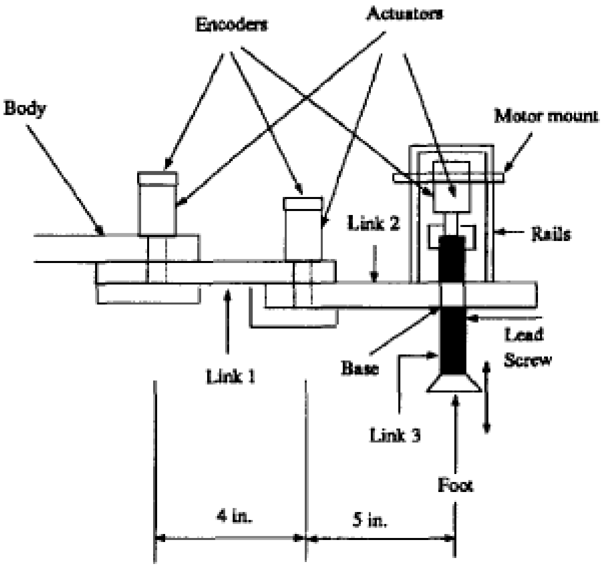


Figure 7 ROACH Leg Mechanism [27]

Each joint of each leg also has an encoded to measure the position of each joint. All of the joints are powered by stepper motors, with power sources connected to the robot by umbilical. The stepper motors are used to reduce power consumption, since while a DC servo motor will increase the voltage and/or current applied to the motor if impeded from reaching its destination angle, a stepper motor will not increase the power consumed.

ROACH utilizes a model based method to identify a staircase utilizing a single image of the staircase. This image is then matched with an existing staircase model through edge detection, and orientation matching. ROACH then uses the computed position of the model to compute its own position, from which it plans a path up the stairs, as shown in Figure 8.

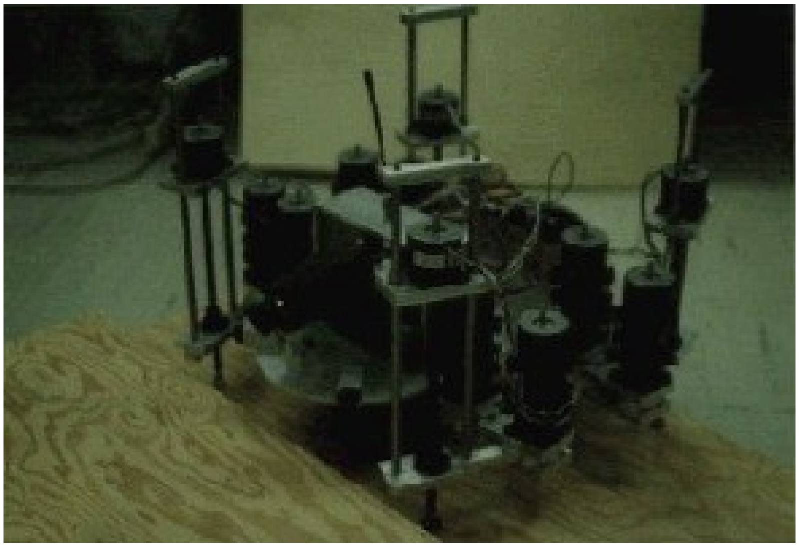


Figure 8 ROACH Quadruped Robot [29]

For planning and gait control, ROACH utilizes a host computer connected to the robot via a serial connection. This host computer is also responsible for the image processing and model matching necessary for stair-climbing.

While being able to recognize stairs by matching them to a pre-existing model, this method is inflexible, in that it requires the computer to already know the height, width and number of the stairs prior to image processing, and therefore ROACH is only capable of climbing stairs already defined by the operator. Also, with the power sources, controller, and imaging system external to the robot, its autonomy is limited.

2.3.3 SCOUT-1/SCOUT II

The SCOUT-1 and SCOUT II robots were developed from 1998-2003 and onwards by researchers at McGill University and Princeton University [30–34]. SCOUT-1 exhibits 4 degrees of freedom, while SCOUT II, a larger version built on the concepts proved by SCOUT-1, utilizes 8 degrees of freedom, though only 4 are actuated.

SCOUT-1 was developed as a low cost, simple design, intended to demonstrate that a robot with limited degrees of freedom could perform feats such as dynamic walking, turning, and step climbing if controlled correctly. It was also discovered that SCOUT could travel utilizing pronking, which involves using all four legs simultaneously to bound forward.

SCOUT-1 consists of four legs, with each actuated by RC servos rotationally around a horizontal axis. This allows independent, controllable motion for each leg. This simple design limits the weight of the robot, making it lighter, however it also limits the payload. This is especially evident in the fact that both the power source and controller for SCOUT-1 were not mounted on the body itself, but are instead connected via umbilical.

SCOUT-1's locomotion is controlled utilizing a static gait for each mode, which is then tuned utilizing a low-level controller based on conservation of momentum. SCOUT-1 utilizes a dynamic bounding gait to climb stairs, as shown in Figure 9. First, SCOUT-1 leans backwards, and then springs forward with both legs leaping on the first stair at once. SCOUT-1 then leans forward and performs the same type of motion with the rear legs to move them forward. This set of motions is repeated several times, until the back legs are close to the stair. SCOUT-1 then leans forward and dynamically rotates both rear legs almost 360 degrees in a rearward motion, with both legs landing on the stair. SCOUT-1 is now on top of the stairs, standing upright.

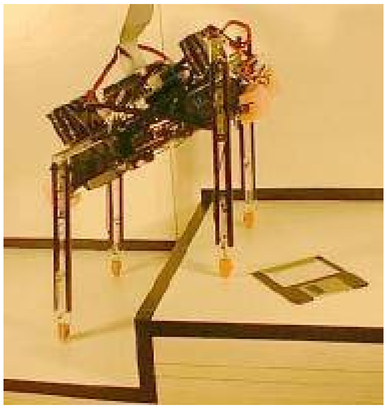


Figure 9 SCOUT Climbing Stairs [35]

While the dynamic gaits demonstrated by SCOUT-1 are impressive, the practical use of the robot itself is minimal, since it must be connected via umbilical to power and control

These issues were addressed with the later development of SCOUT-II. SCOUT-II has the same configuration as SCOUT-1, but with several improvements. SCOUT-II introduced linear spring-damper systems on each leg. This serves to compensate for terrain irregularities and to minimize the shock to the robot itself from dynamic bounding. SCOUT-II also is self contained, with power source, controller, and all required sensors contained within the robot itself. SCOUT-II's instrumentation is more advanced as well, including dual laser range finders positioned at the front and rear of the body, pointed downward. These sensors find the range to the ground of the front and rear of the body, and then use these measurements to calculate the body angle and height. The SCOUT-II, shown in Figure 10, is also capable of climbing stairs, as well as walking, bounding, galloping, and pronking. SCOUT-II climbs stair utilizing the same gait as SCOUT-1.

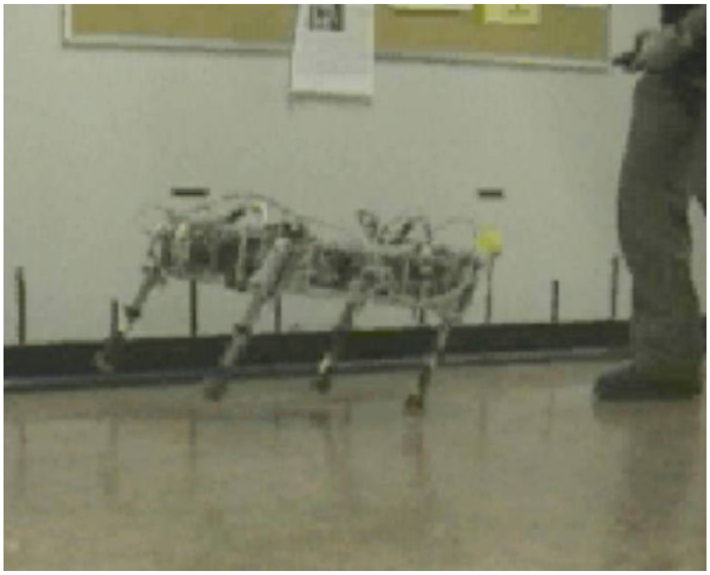


Figure 10 SCOUT-II [36]

While both SCOUT robots are revolutionary in their approach to dynamic gaits and stair climbing, they are both limited in the size of stair that they can traverse. SCOUT-1, with a leg length of 20cm, is only capable of climbing a 9cm stair, while SCOUT-II is capable of climbing 10cm stairs, one third of its leg length.

Also, while self-contained, SCOUT-II has no planning or environment sensing capabilities, and is controlled by an operator utilizing a modified television remote to switch between gaits.

2.3.4 HIT-HYBTOR

Developed by B. Huang et al at the Robotics Research Institute of Harbin Institute of Technology from 2005-2006, HIT-HYBTOR was designed as a hybrid walking and wheeled robot, capable of both quadruped leg-based locomotion as well as wheeled locomotion on smooth terrain [37], [38].

Each of HIT-HYBTOR's legs has three rotational joints, as shown in Figure 11.

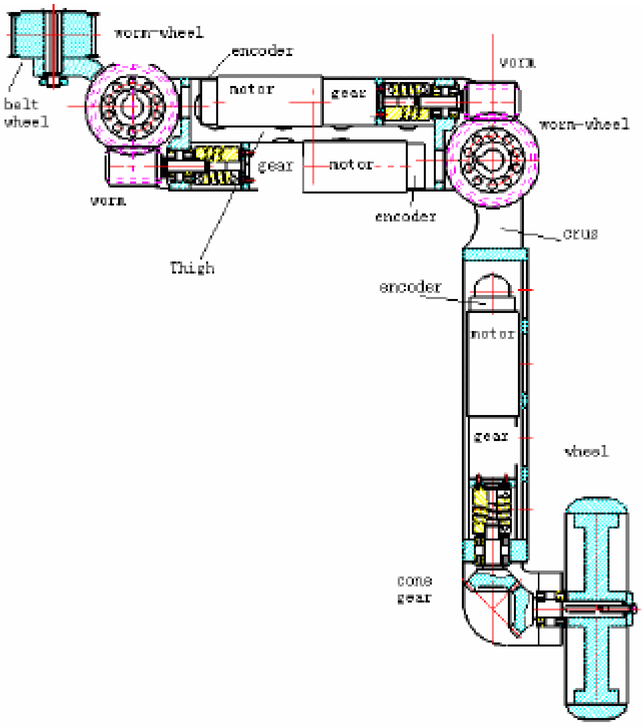


Figure 11 HIT-HYBTOR Leg Configuration [38]

This leg configuration provides a large reachable space, however it limits the available orientation of the wheel/foot, causing the foot to rotate while in contact with the ground, resulting in increased friction.

HIT-HYBTOR is highly instrumented, with joint encoders on all joints, contact and load sensors on all feet, inclinometers for the two horizontal axes mounted on the body, as well as a camera system mounted at the front of the robot. Some of these sensors, such as the camera, have not all been implemented, and therefore are not used for walking or stair climbing.

To control the walking gait, HIT-HYBTOR has an extensive series of “behaviors” that operate at both the leg and body level. The leg-level behaviors perform all leg motion as well as assuring that each leg produces the correct support when in contact with the ground to maintain a stable posture. The body-level behaviors work separately from the leg-level behaviors, and maintain correct body positioning. The leg-level behaviors are run on separate control cards for each leg, and only receive input from the leg sensors. The body-level behaviors receive sensor input from all legs, as well at the body-mounted sensors. HIT-HYBTOR also has a higher-level navigation behavior system, which directs the direction of motion of the robot. Through these behaviors, HIT-HYBTOR is able to maneuver of fairly rugged terrain without extensive prior motion planning or visual sensors.

To climb stairs, as shown in Figure 12, HIT-HYBTOR utilizes its normal walking gait, and the behaviors implemented for the walking gait accommodate the geometry of the stairs automatically. While the maximum stair height that HIT-HYBTOR can climb is not listed, in simulations HIT-HYBTOR is shown climbing stairs nearly half its own leg height.

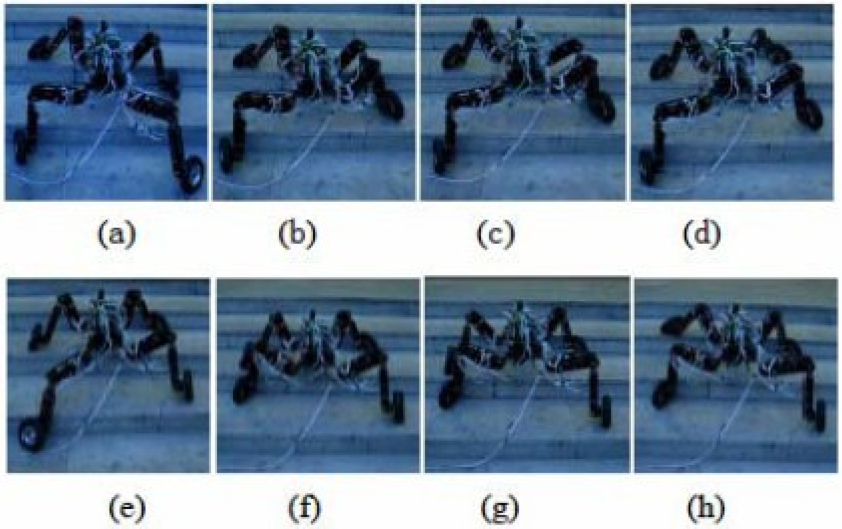


Figure 12 HIT-HYBTOR Climbing Stairs [38]

While a highly capable robot, HIT-HYBTOR is limited by its disuse of available sensors, as well its dependence on an external computer to implement all of its behavior controls. It can climb stairs without utilizing a special gait, and has the added ability of wheeled locomotion on smooth surfaces.

2.3.5 Quattroped

The Quattroped robot is currently in development by researchers at National Taiwan University, with research beginning in 2010 [17–20]. It is a quadruped robot with 8 degrees of freedom, and is similar in configuration to SCOUT-1, though in size it is larger, and it does not utilize a bounding gait to climb stairs. Quattroped has demonstrated the ability to climb stairs with two different types of legs, utilizing both a simple straight leg, actuated with a revolute joint at the shoulder and a linear joint to change the leg length, as well as the same configuration with a hybrid half-circle leg attached to the end of the linear leg, which performs in the same manner as the legs of RHEX[16], with the added benefit of changing the center of rotation with the linear joint. With both leg configurations it is capable of climbing multiple heights of stairs utilizing the same robust gait, including standard stairs of 30-40 degree incline. The Quattroped with hybrid legs is shown in Figure 13.

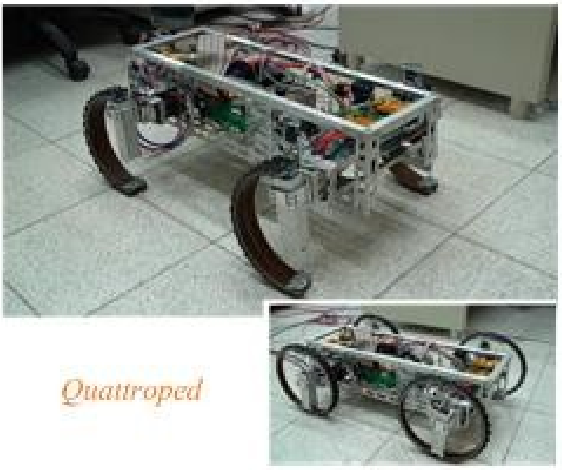


Figure 13 Quattroped with Hybrid Legs [39]

As can be seen above, the “c” shaped legs are formed of two separate pieces mounted on a revolute joint. This allows them to hold the “c” shape for rugged terrain locomotion, and also allows the two pieces to become a circle and the legs are therefore transformed into wheels, which allows the Quattroped to travel quickly on smooth surfaces.

Quattroped’s control system consists of servo encoders on all motors to read their position, an onboard real-time processor which controls all motors and communicates with the encoders. The onboard processor contains all of the control programming and gait information necessary for the robot to operate, however the robot also maintains a wireless connection with an external laptop PC , through which an operator controls the robot by switching between gaits, as well as monitoring the data uploaded, which includes temperature data from all motors, as well as the battery level of the robot.

To execute stair-climbing, the operator positions Quattroped at the bottom of the stair, with the front of the robot facing the stair if the simple straight legs are attached, or with the rear of the robot facing the stairs if using the hybrid legs. The operator then executes the stair-climbing gait, and the Quattroped continues to repeat this gait until the operator sends it a command to stop or begin a different gait. In the figure below, Quattroped is climbing stairs utilizing the hybrid legs, but the gait is similar for both the straight legs and the hybrid legs.

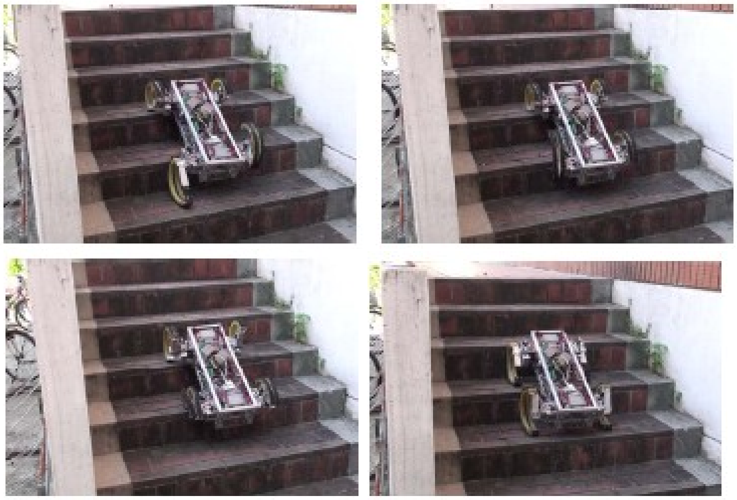


Figure 14 Quattroped Climbing Stairs with Hybrid Legs[20]

Quattroped is capable of traversing a large variety of very rugged terrain, and is also capable of climbing any size of stair. However, it contains no environmental sensors, and must completely rely on the operator to control which gait to use for a particular terrain, as well as provide a course to navigate.

2.3.7 LittleDog

The LittleDog robot was developed by Boston Dynamics in conjunction with DARPA to serve as a platform for the research of machine learning as applied to locomotion, motion planning and gait development [40–50]. Several universities, including MIT, Stanford, Carnegie Mellon, University of Southern California, University of Pennsylvania, and the Florida Institute for Human and Machine Cognition were given LittleDog platforms and specific goals to attempt, such as maintaining a certain average speed over a given square of rough terrain, or to minimize completion time over a given course.

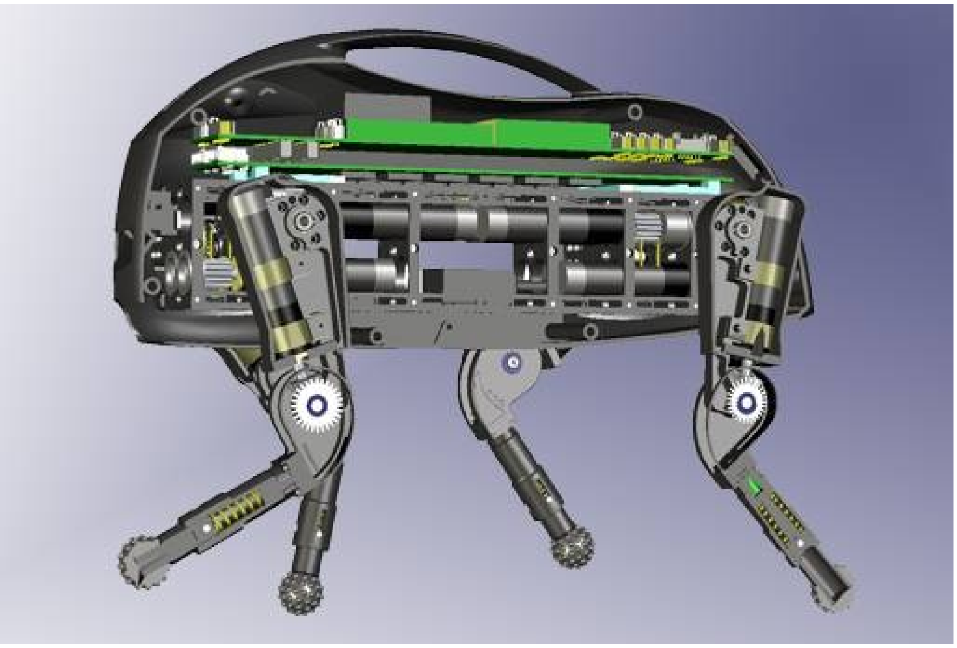


Figure 15 Cross-section View of LittleDog [51]

LittleDog, shown above in Figure 15, has 12 total degrees of freedom, 3 for each leg. Each leg has a rotary lift joint at the shoulder, a swing joint also at the shoulder, and a knee joint further down the leg with a axis of revolution parallel to the swing joint at the shoulder. This joint configuration, along with high torque motors and very high gear ratios, gives LittleDog a very large working space for each foot, throughout which the leg is capable of supporting a large portion of the robot’s weight. This flexibility makes LittleDog an ideal platform for both static and dynamic gaits, as well as stair-climbing.

LittleDog's onboard sensors include 3-axis force sensors on each foot, encoders on each of the 12 joints to measure position, an Internal Measurement Unit (IMU) which measures body orientation, and voltage, current, and temperature sensors, along with a lithium-polymer battery which gives LittleDog 30 minutes of runtime between charges.

The onboard controller for LittleDog is a simple proportional-derivative (PD) controller which regulates and controls joint trajectories. All other controls, including motion planning, calculation of desired joint trajectories, and any other control strategies used, take place on an external computer connected to LittleDog via a 100Hz wireless connection.

This external computer also makes use of a Vicon motion capture system, which provides 6 degree of freedom position information for the robot's body, as well as a millimeter accuracy terrain map of the robot's projected path, with only 50 ms latency.

It has been demonstrated that LittleDog is capable of climbing multiple different sizes of stair utilizing both static and dynamic gaits. K. Byl and R. Tedrake of the Massachusetts Institute of Technology demonstrate the potential of a kinodynamic planning controller which allowed LittleDog to successfully ascend stairs approximately one half the length of the robot's legs in [45]. This bounding motion is shown below, in Figure 16.

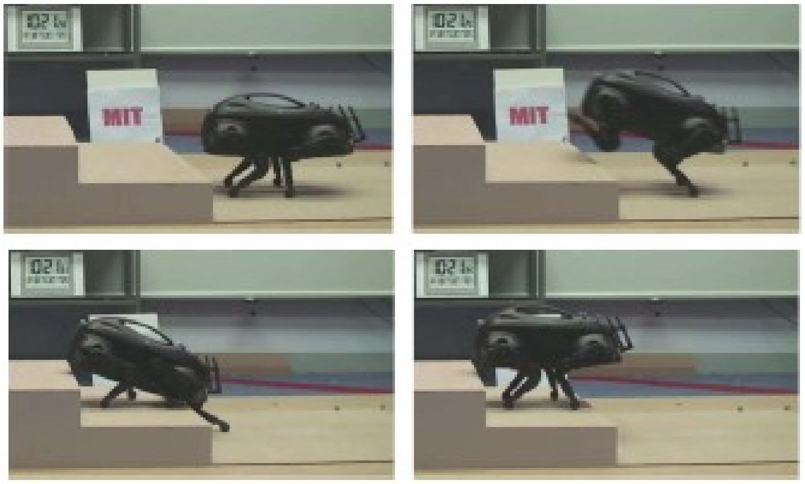


Figure 16 LittleDog Stair-Climbing Utilizing a Dynamic Gait [45]

This gait control architecture can also be applied to rough terrain, as well as smooth surfaces.

Stair-climbing utilizing a static gait has also been developed by researchers at the University of Southern California's Computational Learning and Motor Control Lab (CLMC). This gait is capable of climbing the stairs of the same dimension as the dynamic gait, as well as traversing the same type of rough and smooth terrain. Stair-climbing utilizing the static gait developed at CLMC is shown in Figure 17.

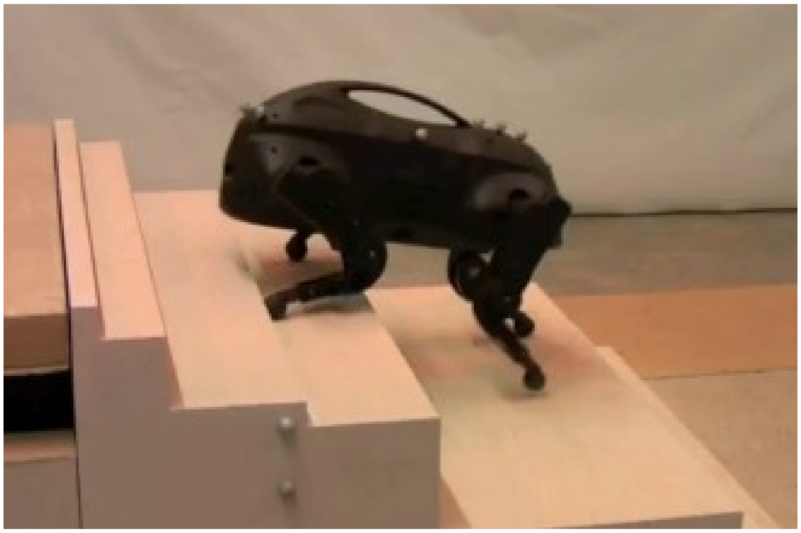


Figure 17 LittleDog Stair-Climbing Utilizing a Static Gait [52]

Overall the LittleDog is an excellent platform, and it has shown the ability to climb any size stairs (up to a kinematics constrained maximum) utilizing a variety of gaits. However, it is constrained by its requirement of both an external computer for almost all control processes, as well as its dependence on the motion capture system to provide it a precise 3 dimensional terrain image, as well as its own position.

2.4 Vision and Sensor Based Stair Detection

In order to climb stairs, a mobile robot must have some way of detecting the stairs and obtaining parameters corresponding to that particular set of stairs. This sensing can take many forms, from as simple as having a human operator recognize the stairs and direct the robot to execute the correct action or series of action necessary to climb them, to as complicated as developing an entire 3 dimensional model of the stairs and planning a complete path for the robot to follow. For many of the robots discussed thus far, both quadruped and others, this stair detection problem is just as difficult to overcome as the calculations and controls to get the robot to physically climb the stairs. In the following discussion we will discuss the vision based techniques, as well as those involving other range-based methods. Since the scope of this thesis does not include utilizing touch-based sensors to detect stairs, we will neglect a discussion of these methods.

2.4.1 Stereo Vision Based Detection

One of the more obvious methods to detect stairs is through the use of Stereo Vision. Humans use Stereo vision to recognize objects as well as to determine distance, and it is therefore logical to utilize the same technique for mobile robot sensing, particularly in the case of stair recognition.

Researchers at the Intelligent Systems Research Laboratory in Tokyo have developed a system through which a bipedal robot utilizes stereo vision to recognize and then climb and descend a set of stairs [53]. To recognize the stairs, the QRIO robot first receives stereo image data, including range information, from its camera array. This depth information is then processed, and through the use of plane segmentation each stair is identified, from which the entire set of stairs is computed. This information is then used to allow the robot to climb or descend the stairs.

Figure 18 QRIO Robot [54]

A variation of pure stereo vision is demonstrated by researchers at the University of Hannover, with the use of stereo vision combined with dual line lasers to detect stair dimensions [55]. Their bipedal robot BARt-UH projects two separate line lasers onto the surface of the stair, one for each camera. A threshold operation is then performed on the images captured by each camera to create binary images. Next, stereo triangulation is performed to find the corner points in the line made by the laser, with the corner points corresponding to the edges of the stair. The two stereo images are then compared and the dimensions of the stairs are computed, allowing the robot to climb the stairs. A sample of the images captured by the vision system, as well as the images post-processing are shown in below.

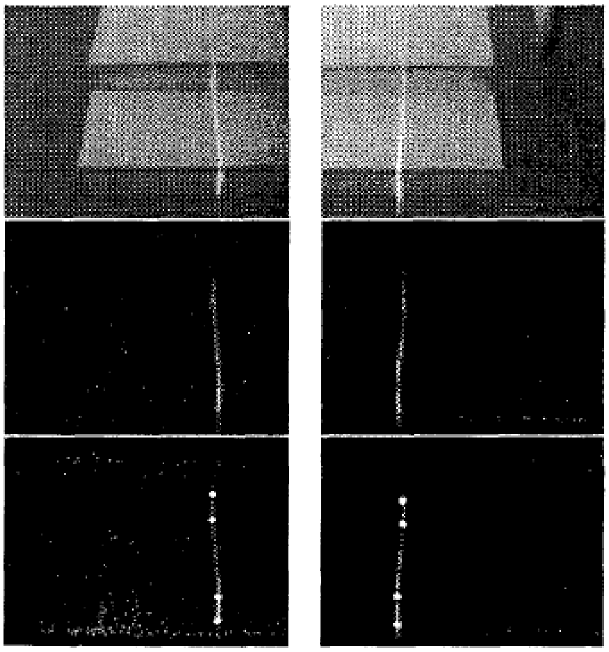


Figure 19 Stereo Vision Recorded by BARt-UH [55]

One disadvantage of stereo vision is the computing power required to process two different images simultaneously and then compare them and derive depth information. With the recent increases in computing power and reduction in size and weight required to support those capabilities, stereo vision is becoming more feasible, and will most likely be used increasingly for autonomous robots.

2.4.2 Range-Finder Based Detection

Another method for stair detection is to utilize a range-finding sensor and use the distance information found to then extrapolate stair dimension information. While efficient at stair characteristic determination, this method is less useful for general navigational purposes due to its small detection are, and therefore not as popular. However, researchers at Osaka University have developed a hexapod robot ASTERISK which utilizes a laser range finder mounted on a motor to develop a 3 dimensional picture of its surroundings [15]. This process involved completing 360 vertical line scans of the surroundings, then identifying stairs through line detection, and finally building a motion plan to climb the detected stairs. A comparison of the robot's environment and the information generated from the range-finder scan are shown in Figure 20.

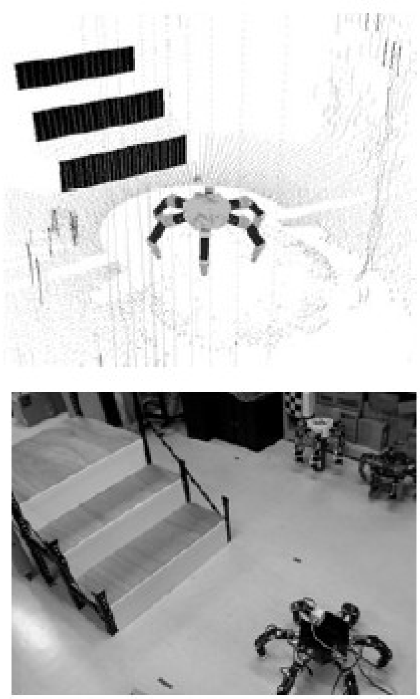


Figure 20 Results of ASTERISK's Range-Finder Scan [15]

While the data for the environment is collected, the robot must remain stationary, and due to the inefficient nature of data collection, ASTERISK may spend a large amount of time scanning parts of the environment which do not contain stairs. However, the resolution and accuracy of this method is high, which allows for efficient stair-climbing by ASTERISK once the stairs have been detected.

2.4.3 Image Processing Based Detection

Image processing for stair detection is the most common method in use by mobile robots today. However, there are several methods for stair detection utilizing image processing.

In the Edge-detection method, analysis is performed on images from a single camera, and through utilizing filters, transforms, and search techniques, the lines which form the edges of the stairs are found. With these, the dimensions and location of the stairs can be found, allowing the robot to climb the stairs.

To allow their Tactical Mobile Robot (TMR) to climb stairs, researchers at the Jet Propulsion Laboratory have utilized the edge detection method to detect and analyze stair dimensions and configuration, which has allowed the TMR to climb stairs autonomously [56]. Their approach is slightly different than standard edge detection, in that they do not utilize a threshold filter initially, and instead take a least commitment approach to preserve stair edges. This increases the spurious edges detected, but this is compensated for by utilizing a search algorithm with finds long, mainly horizontal, parallel edges, and then utilizes these remaining edges to calculate stair characteristics. The result of this process is shown in Figure 21, with the real-world images on the left, and the resulting stair edges on the right.

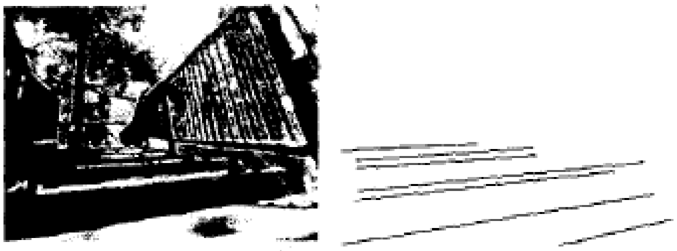


Figure 21 TMR Stair Detection [56]

Another type of image processing is the use of single camera Motion Stereo vision. Essentially, this involves utilizing a single camera mounted on a moving platform to collect different perspectives of an object, and by knowing the position of the camera at for each image used, the location of the object can be found, as well as the shape and characteristics of this object. This is well suited for some types of robot, such as tall bipedal or flying robots, since they are highly mobile and can easily achieve several different perspectives on an object. For smaller, fixed height robots, this method can prove problematic, since by having a viewpoint fixed to the robot, they are limited in the number of perspectives they can attain on a particular object. This is especially true of stair detection, since actual staircases are contained in small spaces, which limits the perspectives available, particularly for small robots.

Two robots which demonstrate effective single camera motion stereo vision are the previously mentioned HRP-2[5], and the Johnnie robot used by the Institute of Automatic Control Engineering, Technische Universitat Munchen [56]. Both of these robots are tall, bipedal robots which utilize their mobility to gain multiple perspectives on objects including stairs, which then allowed them to ascend the stairs.

2.4.4 Microsoft Kinect in Robotics

As one of the newest commercially available sensors for use in robotics, the potential of the Microsoft Kinect has yet to be fully realized in its application to mobile robots. Some research has been published demonstrating the use of the Kinect, and its use is very wide-spread, which is anticipated to lead to a large increase in Kinect related work.

One use of the Kinect thus far has been to utilize the depth map information generated by the Kinect to assist in the altitude control of a quadrotor helicopter [57]. Researchers at the University of Canterbury in Nez Zealand have developed a calibration and alignment technique to ensure accurate depth readings from the Kinect, which they then attached to the underside of their quadrotor. With the Kinect pointing at the ground, they were able to determine the distance from the quadrotor to the ground, and were thus able to control the quadrotor's altitude with a PID controller using this information, as well as input from other sensors. While a very simple use of the Kinect's capabilities, it nonetheless shows that a Kinect can be mounted on a mobile robot and used successfully as a sensor for control-based navigation.

Another recent use of the Kinect system has been the development of a control architecture which utilizes the Kinect's sensor data within a larger control system to perform navigation and target tracking for a tracked mobile robot [58]. Researchers at the University of Texas, San Antonio have proposed a system which will derive a terrain model from the depth and RGB color information provided by the Kinect. This would allow a mobile robot to sense all of the terrain around it and plan an appropriate path to reach any objective.

Chapter 3 – Design of Quadruped Robot

3.1 Overview

In this chapter, the design, modeling, and implementation of a self-contained stair-climbing quadruped robot will be discussed. The design, manufacture, and validation modeling are fully described.

In Design Concept, we will discuss the development of the unique leg-joint configuration and how this design is incorporated into the design of the robot as a whole.

In Selection of Commercial and Off-the-Shelf Components the selection of all the commercially available components used to construct the robot will be discussed. In Design of Custom Components the selection of the materials from which to construct the robot will be discussed, as well as several unique mechanisms which greatly improve the performance of the robot.

In Simulation-Based Sizing of Components, the use of computer modeling to determine the component sizes used in the robot is discussed. A dynamic simulation of the robot performing the most difficult stair-climbing motion will be examined, and a Finite Element Analysis will be performed on the component of the robot at highest risk of failure. Both of these simulations will serve to validate the robot design prior to construction.

3.2 Design Requirements and Specifications

Various approaches to stair climbing have been previously examined. While most of the designs contain some onboard sensors, those with a large sensing capacity or extensive gait generation and planning capabilities usually have the majority of these sensors mounted externally, not on the robot itself. This is nominally because of the increased payload capacity sensors and complex controllers require. These increased weight requirements lead to a need for larger motors or actuators, which in turn again increases the robot's weight, which leads to higher capacity motors, etc.

Therefore, the main design goal for this robot is to create a platform which is capable of transporting itself up stairs of multiple dimensions, while also being capable of carrying both the sensors and controllers capable of sensing the stairs and guiding the robot to climb them.

Another goal is to manufacture the robot utilizing materials and components readily available commercially. While this may hamper optimization efforts, since commercially obtained materials may not be perfectly suited for a particular task, it will decrease the cost of the robot. By using commercially produced components, some complexity will be reduced as well, since the ability to utilize modularity will be increased. This goal extends to the software used to control the robot and the sensors as well. An effort will be made to utilize readily available software to control the robot. This will also decrease the cost of the robot, as well as reduce the amount of time needed to develop the controls themselves, since much can be based off of open source software.

The robot also needs to be capable of walking on flat terrain as well as stair-climbing. Many stair-climbing specific robots have been developed and analyzed, but the concepts used have less importance to real-world applications. The intent in this project is to create a robot that, while capable of self-contained stair-climbing, is also capable of locomotion over other types of terrain. The development of gaits and controls to enable the robot to traverse any terrain besides stairs and simple flat planes will not by discussed in this effort, though it will be discussed in the Future Work chapter.

Therefore, the overall goal of this design is to construct the lightest possible robot with 12 active degrees of freedom and 4 passive degrees of freedom, capable of climbing stairs while carrying all sensors and controllers required to detect stairs and guide the robot to climb them. It must also be capable of climbing all stairs with step heights between 2.5 and 5 inches, and therefore it must be able to detect the size of the stair as well.

3.3 Design Concept

The design of this system was completed using Pro Engineer 5/CREO (PRO/E). This includes the three-dimensional modeling, component assembly, finite element analysis (FEA), and dynamic modeling. PRO/E was used for its great flexibility in component design, and for its proven reliability and validated FEA and dynamic modeling.

As a basis for the design of the robot, several assumptions and decisions must be made. While there are several approaches to stair-climbing, one must be chosen for the robot to be optimized for this method. Some robots, such as LittleDog[40–50], [59] utilize a method in which the robot transports its entire body up each step one at a time, with the body of the robot resting on up to two steps at a time. Other robots, such as Quattroped [17–20] span multiple steps simultaneously. In this design, only one step at a time will be ascended, therefore the legs and body must be compact enough to allow the robot to fit its entire body and legs on one stair at a time. Minimum stair width in International Building Code (IBC) is 11 inches, and the minimum height is 4 inches [60]. Therefore the robot will be designed to climb stairs of at least 5 inches in height and a minimum of 10 inches in width. The one inch difference between the designed robot's capability and the minimum IBC height and width is to allow for a margin of error in both the robots motions and sensing capability.

With these stair dimensions, a climbing technique similar to LittleDog's will be utilized, where the entire robot body is moved one step at a time, with the legs of the robot only contacting up to two adjacent stairs at a time.

Also, it is assumed that the surface of both the ground and the stairs will be smooth and flat. While stairs are constructed as well as covered with many different types of material, this simplification is to narrow the focus of the design effort, and in the future other surface materials may be investigated.

3.3.1 Joint Configuration

The joint configuration is the basis for success of any robot, and in the case of stair-climbing, joint configuration determines the required torque at each joint, as well as the configuration of the robot as a whole. For this reason, the joint configuration for the legs of the robot will be discussed first.

For a stair-climbing robot, vertical lifting capacity is highly important, since the robot will be required to lift itself vertically up stairs. This involves high torque requirements, and it dictates a balance between leg length and range of motion. While longer legs provide a larger working space for the leg, it also linearly increase the torque required to move the joint through an angle while under load.

It was for these reasons that the joint configuration shown in Figure 22 was chosen for the robot. This joint configuration allows for three rotational degrees of freedom and maximizes the working space of the robot.

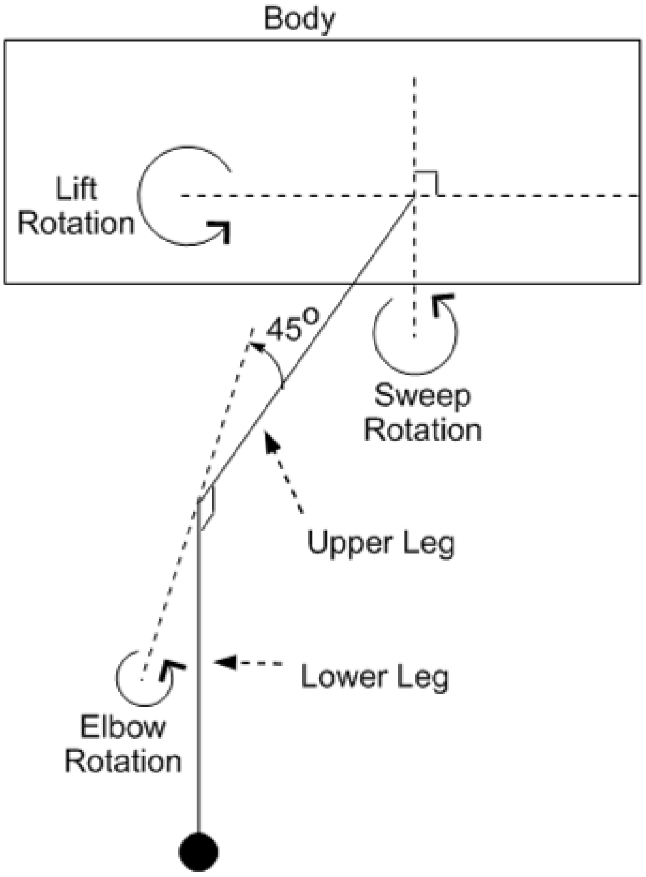


Figure 22 Robot Leg Joint Configuration

The first rotational joint is attached to the body, and rotates around an axis in the plain of the ground, running between the front and back of the robot. Rotation around this axis serves to lift the rest of the leg assembly, and therefore this joint will be referred to as the Lift Joint.

The second rotational joint is approximately co-located with the lift joint, but it rotates about an axis perpendicular to both the plain of the ground and the lift axis. This vertical axis allows the leg to rotate the leg forward and backward in relation to the body, and will be referred to as the Sweep Joint.

The third rotational joint is connected to the sweep joint by the first leg segment. Its axis of rotation is rotated 45 degrees from the direction of the sweep axis in the plain perpendicular to the lift axis. This angle or rotation allows for an increase in the working space of the leg, without increasing leg length. This joint will be referred to as the Elbow Joint.

The final component of the leg is the lower leg and foot assembly. This second leg segment is attached to the elbow joint at a 45 degree angle to the joint, which aligns it with the sweep axis when the leg is at its median position. This combination of an angled axis of revolution and the leg's angled attachment to the axis of revolution is key to the leg's ability to provide not only a large working space with less torque required, but also to allow for a smooth walking gait, which is essential particularly for vision-based sensors.

The final configuration item to discuss is the leg length. This is related to the amount of torque available for each joint, as well as the weight of the robot itself, and therefore will be discuss in a later section.

3.3.2 Robot Configuration

With the leg configuration as described above, the full configuration of the robot can be described. With the need to accommodate both sensors and controllers, the body is required to be large enough to provide the space necessary, while also not being too large to climb the previously defined minimum stair dimensions. With the climbing strategy previously described, a body length of 14 inches was selected. While larger than the intended step, this size is still capable of maintaining its entire body on one step at a time, and is also large enough to accommodate all required controllers and sensors. The body width is determined to be 10.5 inches, which was determined by the required leg and actuator configuration. The legs are attached to the body as shown below, in Figure 23.

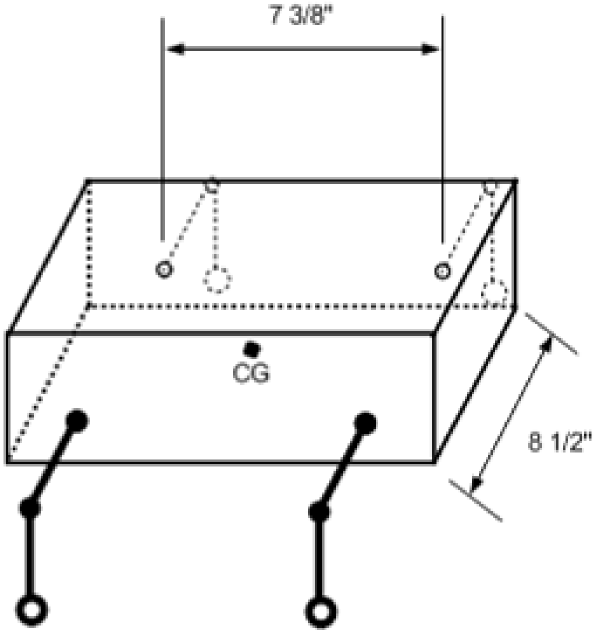


Figure 23 Leg and Body Configuration

This configuration has each of the legs attached at a corner of the robot body, with two legs to each side, and the front half a mirror of the back half, and the left side a mirror of the right. This symmetry allows for equal distribution of the weight associated with the actuators, with any other needed weight balancing accomplished through placement of the other components, such as the battery, controllers and sensors.

3.4 Selection of Commercial-Off-the-Shelf Components

3.4.1 Actuators

The selection of actuators will be discussed first, since this component determines the required specifications of many of the other components. To maintain the simplicity and modularity of the design, the same type of actuator will be used for each joint. With each joint being a revolute joint, a rotational actuator is the obvious choice for utilization.

Many different types of rotational actuators are commercially available, and all have their advantages and disadvantages. While efficient and powerful, pneumatic actuators require a constant source of compressed air. Both compressors and compressed air tanks are heavy, however, and would limit the mobility of a robot of this size. Hydraulic actuators are also very powerful, but the weight issues encountered are similar to pneumatic actuators.

The most common actuator utilized for small mobile robots are Direct Current (DC) motors, due to their good power to weight ratios, as well the availability of high power batteries. For this application, DC servo motors are the most useful. While it may be possible to use a simple rotational DC motor implemented utilizing a high gear ratio to produce the required toque, servo motors perform the same function using the same means, however this entire mechanism is packaged in a small, compact volume, along with electronics which control the system and can move the servo head precisely. An illustration of a servo motor is shown in Figure 24.

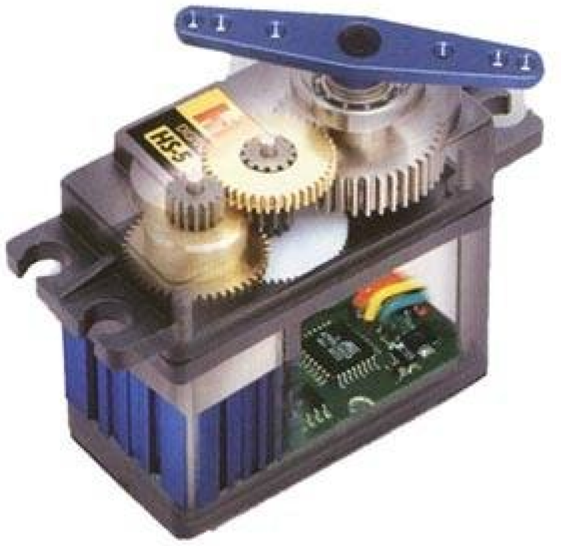


Figure 24 Servo Motor Cut-away View [61]

While there are many different sizes and configurations of servo motors, most of them fall into three main categories: Analog Control, Digital Control, and Stepper. Analog servos are controlled utilizing Pulse Width Modulation (PWM). PWM servos are controlled by applying an input voltage on and off very frequently, with the ratio of time the voltage is applied to the time the voltage is off controlling the position of the servo. For most analog servos used in robotics, the step time is 0.2 seconds, or in other words the sum of the time the voltage is applied and the time the voltage is off is always 0.2 seconds, which means a new control input can be input to the servo every 0.2 seconds. The range of input time length varies from servo to servo, but is typically 900 to 2100 micro-seconds. The advantage to this type of servo is that its power consumption is moderate, with the input voltage only being applied for a fraction of a second. A disadvantage is that with the lower power input, the torque generated is also nominally lower than other types of servos. The highest torque available in analog servos is typically 300-350 oz-in, with a corresponding weight of 4.5-6 oz per servo.

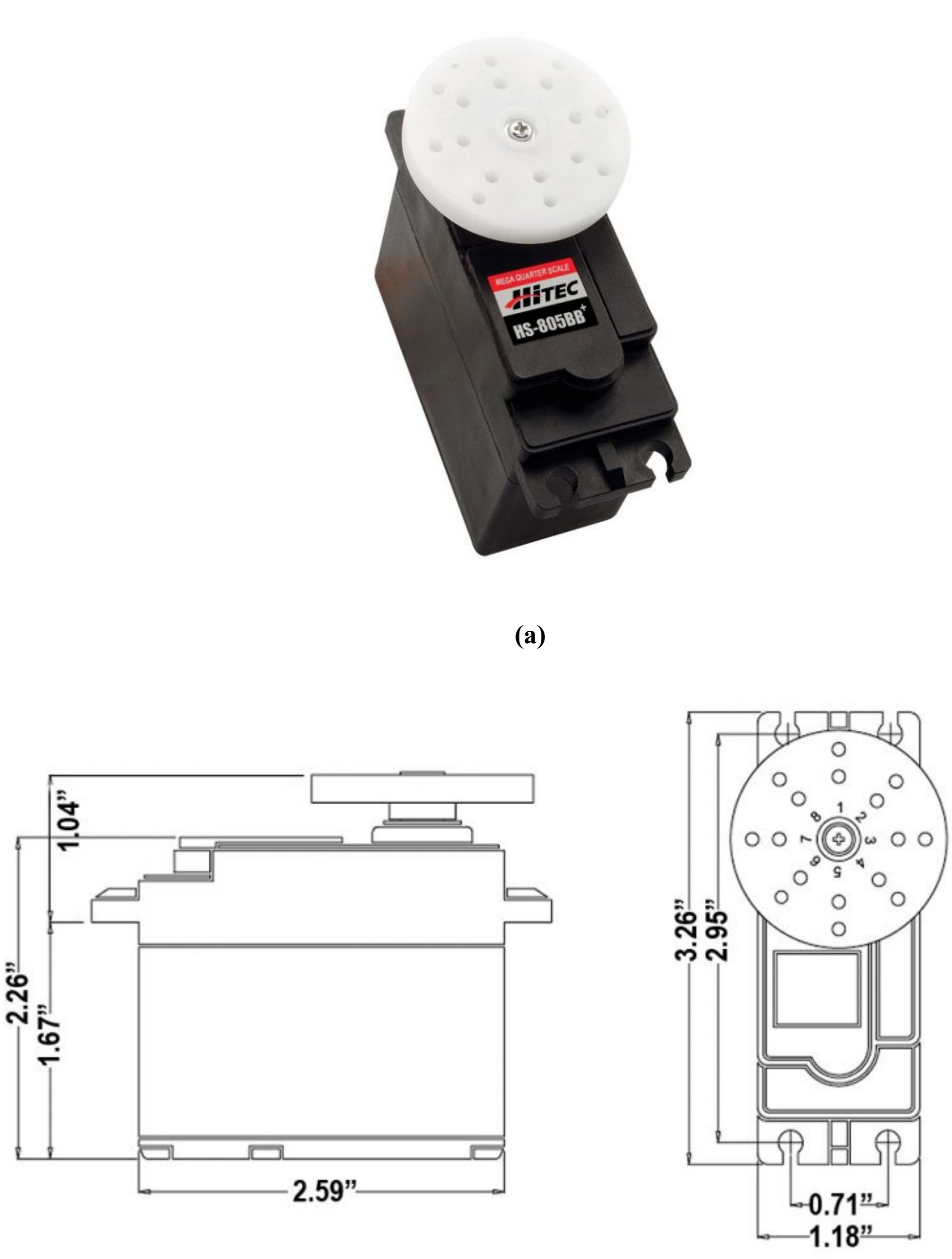
Digital servos are essentially the same device as an analog servo, with the exception of the internal controller. The digital internal controller receives a digital signal directing the servo to turn to a particular angle, and it receives this signal at a much greater frequency than an analog servo. A digital servo will receive a signal with a frequency of over 6 times that of a comparable analog servo. This enables the digital servo to respond to input much faster, and with much greater torque, than an analog servo. However, with this faster response time and increased initial torque comes a greater power drain. Because of this, digital servos require higher power batteries, which they will discharge in less time, compared to analog servos.

Finally, Stepper servo motors have a different mechanism than both analog and digital servos. Rather than utilizing a DC electric motor, steppers contain a brushless electric motor made of a single center gear-shaped pieced of ferrous metal, surrounded by multiple toothed electromagnets. These electromagnets are energized in sequence, which causes the center piece to rotate. This configuration allows for very precise control of the position of the motor, as well as very high output torque.

However due to the solid center piece and the multiple electromagnets, stepper motors are much heavier than digital or analog servos which give comparable torque. For this robot, analog servos were utilized for each revolute joint, due to the smaller required input power and lighter servo weight. Because high torque is required for each of the joints, a high torque HiTec HS-805BB servo was used. The specifications for this servo are shown in Table 1, and the servo itself is shown in Figure 25, along with a dimensioned drawing.

Table 1 HiTec HS-805BB Analog Servo Specifications [62]

Motor Type:	3 Pole
Bearing Type:	Dual Ball Bearing
Speed (Sec/60 degree rotation):	0.14
Torque (oz/in):	343
Size (inch):	2.59 x 1.18 x 2.26
Weight (oz):	5.40
Operating Voltage:	6V
Stall Current:	4A



(a)

(b)

Figure 25 (a) HiTec HS-805BB Servo (b) Dimensioned Drawing of HS-805BB [62]

This servo has one of the highest maximum torque to weight ratios available, and the HiTec brand is one of the most commonly used, and one of the most reliable. Also, at a cost of approximately $40 per servo, it is one of the cheaper servos available.

3.4.2 Sensors

The next components of the robot which will be discussed are the sensors. The sensors must be capable of providing the information required to a controller to allow the controller to recognize the stairs, as well as determine the location of the stairs in relation to the location of the robot. While there are a large number of commercially available sensors, the large majority of them only provide part of the information required. For this reason the Microsoft Kinect sensor is used as the main sensor on the robot. The Kinect is capable of providing not only image information but depth information for that image as well. This means that with the correct implementation a Kinect Sensor could provide a three-dimensional terrain map to a controller. The Kinect is comprised of several sensors, including a color camera, an infrared camera, an infrared emitter, and four separate microphones. The Kinect's exact specifications are provided in Table 2, and the Kinect sensor itself is shown in Figure 26.

Table 2 Microsoft Kinect Sensor Specifications [63]

Color Camera Resolution	640x480, 32 bit
Color Cameral Frames Per Second	30
Infrared Camera Resolution	320x240, 16 bit
Infrared Camera Frames Per Second	32
Vertical Field of View	43 Degrees
Horizontal Field of View	57 Degrees
Minimum Depth Sensor Range	4 Feet
Maximum Depth Range	11.5 Feet
Voltage Input Required	12 Volts
Power Input Required	12 Watts
Weight	2.25 lbs

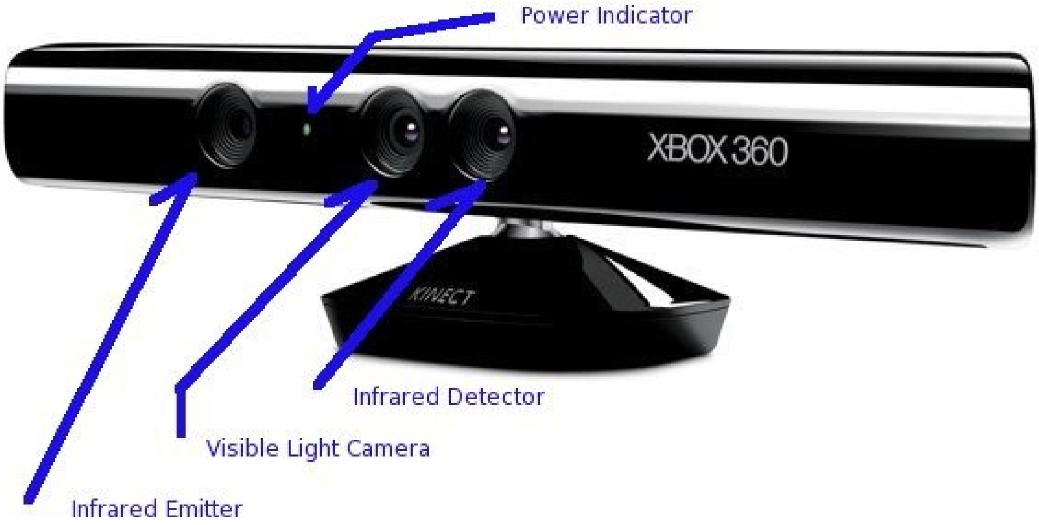


Figure 26 Microsoft Kinect Sensor [63]

The Kinect is powered by a proprietary connector which is normally connected to an Xbox 360 console, however since a different controller will be used, a different connection scheme is required. An external power source cable is sold with the Kinect system, since older model Xbox consoles do not have the required connection. This power source cable has a female connector which connects to the Kinect, which then splits into a standard USB 2.0 connector and a 12 Watt DC voltage inverter, which then plugs into a standard 120V AC wall socket. Since it is necessary to power the Kinect using a power source carried onboard the robot, an alternate power source is required.

For this power source a Tenergy NiMH 12 Volt 2000 mAh battery pack is used. This rechargeable battery pack is capable of providing over 1 Amp of current at 12 Volts for over an hour, and weights only 10 oz [64]. This battery is therefore capable of providing power to the Kinect sensor at 12 Watts.

However, to protect the Kinect sensor from possible power irregularities, a 12 Volt Voltage regulator was constructed using the schematic shown in Figure 27.

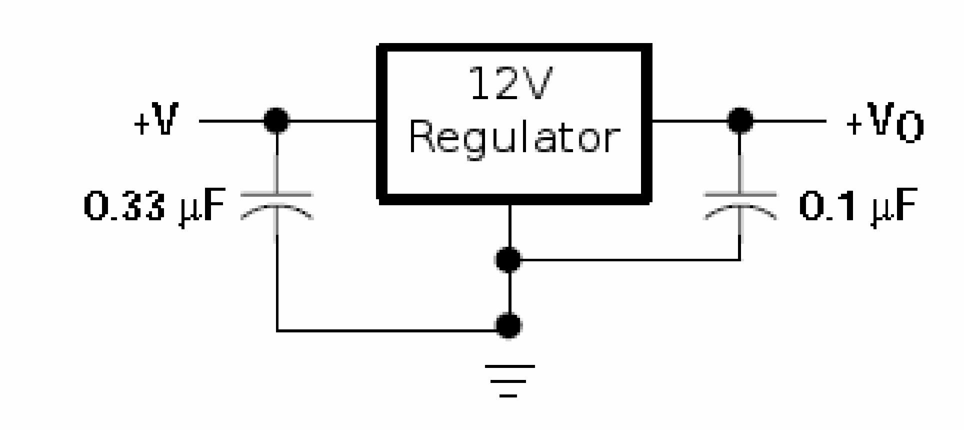


Figure 27 Voltage Regulator Schematic [65]

This voltage regulator was then connected to the 12 volt battery and the Kinect's male proprietary connecter, providing power to the Kinect while freeing the USB connection to be connected to a controller. The completed voltage regulator and Kinect power cable are shown below in Figure 28.

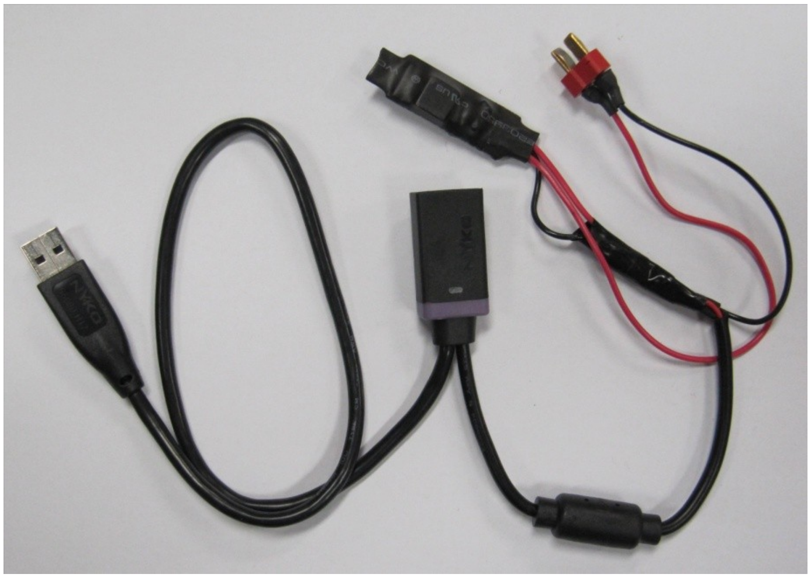


Figure 28 Completed Voltage Regulator and Kinect Power Cable

In Figure 28, in the center is the female proprietary Kinect plug, with the standard USB connection on the left, which connects to the computer. The voltage regulator itself is seen at the top of the figure, with the red battery connection to its right.

One other sensor type utilized is a simple infrared distance sensor, used to determine when the robot is in close proximity to a step. Two HiTec H7 Infrared Distance sensors are utilized on the robot. The sensors have an operating voltage of 5 Volts provided through the Arduino controller, and have a rated detection range of approximately 15 cm, however from experimentation it has been determined that the actual range is approximately 2-3 inches [66]. These sensors have a simple infrared transmitter and receiver assembly which emits an IR signal, which then reflects off of surfaces directly in front of the sensor. This reflected signal returns to the receiver, which then transforms the signal into a step voltage increase when the reflected IR transmission has a large enough intensity, which occurs when an object lies in range of the sensor. The H7 sensor is shown in Figure 29.

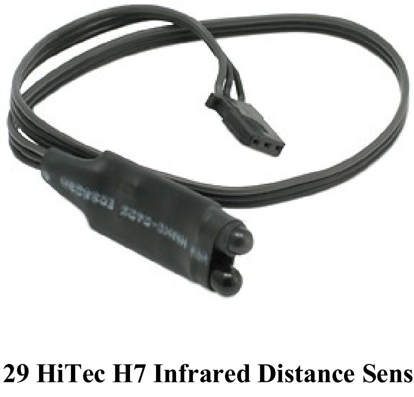


Figure 29 HiTec H7 Infrared Distance Sensor [66]

3.4.3 Batteries

The 12 servos will be powered by two separate battery packs, six servos to a battery pack, to reduce the amount of peak current required from each pack. The two battery packs utilized to power the servos are Nickel-Metal Hydride (NiMH) 6 Volt 5000 milli-Amp-Hours (mAh) rechargeable batteries constructed from 5 Tenergy C-cell 1.2 volt NiMH 5000 mAh batteries connected in series [67]. A single C-cell battery is shown in Figure 30.

Figure 30 1.2 Volt C-Cell Tenergy Battery[67]

Each of these smaller batteries has a weight of 3 oz, which puts the total weight for both servo power battery packs at 30 oz. The c-cell batteries have a C rating of up to 5C, which means that each cell is capable of discharging at up to 5 times the capacity rating. This C rating allows each cell to discharge at up to 25 Amps for up to 9.6 minutes, therefore each battery pack is capable of supplying up to 25 Amps of current, which is greater than the peak current requirement of 24 Amps, which occurs if all 6 servos attached to a battery pack reach stall current at the same time.

3.4.4 Controllers

To control the servos, an Arduino Mega 2560 microcontroller board, manufactured by Arduino, which manufactures several different types of microcontrollers and distributes the required programming software through a free download from their website. The specifications of the Mega 2560 are shown in Table 3, and the controller itself is shown in Figure 31.

Table 3 Arduino Mega 2560 Specifications [68]

Operating Voltage	5V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (14 PWM Output Capable)
Analog Input Pins	16
DC Current per I/O Pin	40mA
Flash Memory	256KB (8KB Used for Bootloader)
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

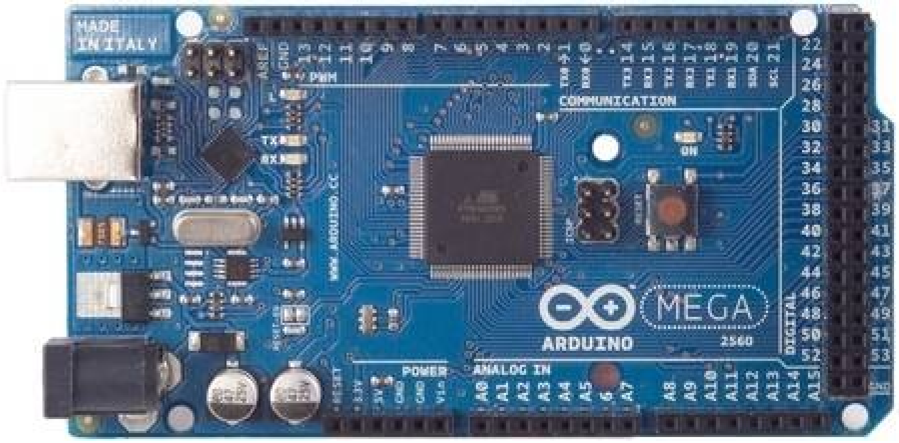


Figure 31 Arduino Mega 2560 Microcontroller [68]

The Mega is capable of receiving power from either a voltage input or the USB connection. The USB connection serves to program the controller, and can act as a serial input as well. A very flexible controller, the Mega is being utilized for the robot due to its high number of PWM output pins, which allows all 12 PWM servos to be controlled by the Mega, which makes servo control very easy to implement, since there is no coordination between controllers or reliance on precise timing synchronization between controllers for separate servos or legs of the robot.

The next component of the robot to be discussed is the main controller. This controller needs to be lightweight, but capable of performing the required image processing of images and data from the Kinect, as well as interact with the Arduino Mega servo controller. A Macbook Air was chosen as the controller due to its low weight and ability to run the software needed for the Kinect. Specifications for the Macbook Air used are shown in Table 4.

Table 4 13-in Macbook Air Specifications [69]

Weight	2.96 lbs
Size	0.68 x 12.8 x 8.94 in
Processor	1.7 GHz Dual-Core Intel Core i5
Memory	4 GM RAM, 128 GB Flash Storage
Ports	Two USB 2.0

The Macbook Air utilized the Windows 7 operating system, along with other software which will be discussed in a later chapter.

3.5 Design of Custom Components

The main construction material for the body and legs used for the robot is required to be low-cost and lightweight as well as strong; however it also should be easily shaped. The material chosen for the construction of the robot is Delrin Acetal Resin, a thermoplastic polymer that has high tensile strength, is commercially available in many different shapes and sizes, is relatively inexpensive, and has high stiffness. The characteristics of Delrin are shown in Table 5, and Delrin sheets like those used for the construction of the robot are shown in Figure 32.

Table 5 Delrin Acetal Resin Material Properties [70]

Tensile Strength	10,000 psi
Impact Strength	2.3 ft.-lbs./in
Hardness	Rockwell R: 120
Opaque	Black or White

Figure 32 Delrin Acetal Resin in Sheet Form [71]

As previously described, the robot components to be made from Delrin were designed utilizing Pro/E. These components were then manufactured utilizing a Universal Laser Systems VLS3.60 laser cutter equipped with a 60 watt laser [72]. The VLS3.60 has a 12”x24” working area, and is capable of cutting through up to ¼” thick sheets of Delrin with an accuracy of approximately one-hundredth of an inch, depending on the settings used. This system allowed for high precision manufacturing of Delrin parts which promoted a good fit between connected parts.

Because of the material thickness restriction on the laser cutter, as well as to reduce weight as much as possible, all Delrin components were manufactured from sheets of 1/8” and 1/4” inch thick Delrin sheets.

While the previously discussed HS-805BB servos are utilized to actuate all of the rotary joints of the robot, to reduce unwanted transverse torque on the servo’s rotational joints, all of the robot’s rotational joints were reinforced with non-actuated joints opposite to the servo’s actuated joints. These joints are constructed using simple 1/8” diameter brass rods rotating inside brass tubes with an outer diameter of 1/4”. These simple joints were used due to their ease of manufacture, their low friction coefficients, the low cost of material, and their low weight.

In the construction of the robot, several unique mechanisms were utilized to increase the performance of the robot. The first mechanism used consists of a linkage system used to connect the lift actuator with the lift joint of the robot. By connecting the actuator’s rotating servo arm to the linkage rod one inch from the center of rotation, and connecting the opposite end of the linkage arm to the rotating lift joint two inches from the center of rotation, the effective torque of the servo on the joint is double, while decreasing the usable rotation space of the lift joint. A ProE model of this linkage is shown in Figure 33.

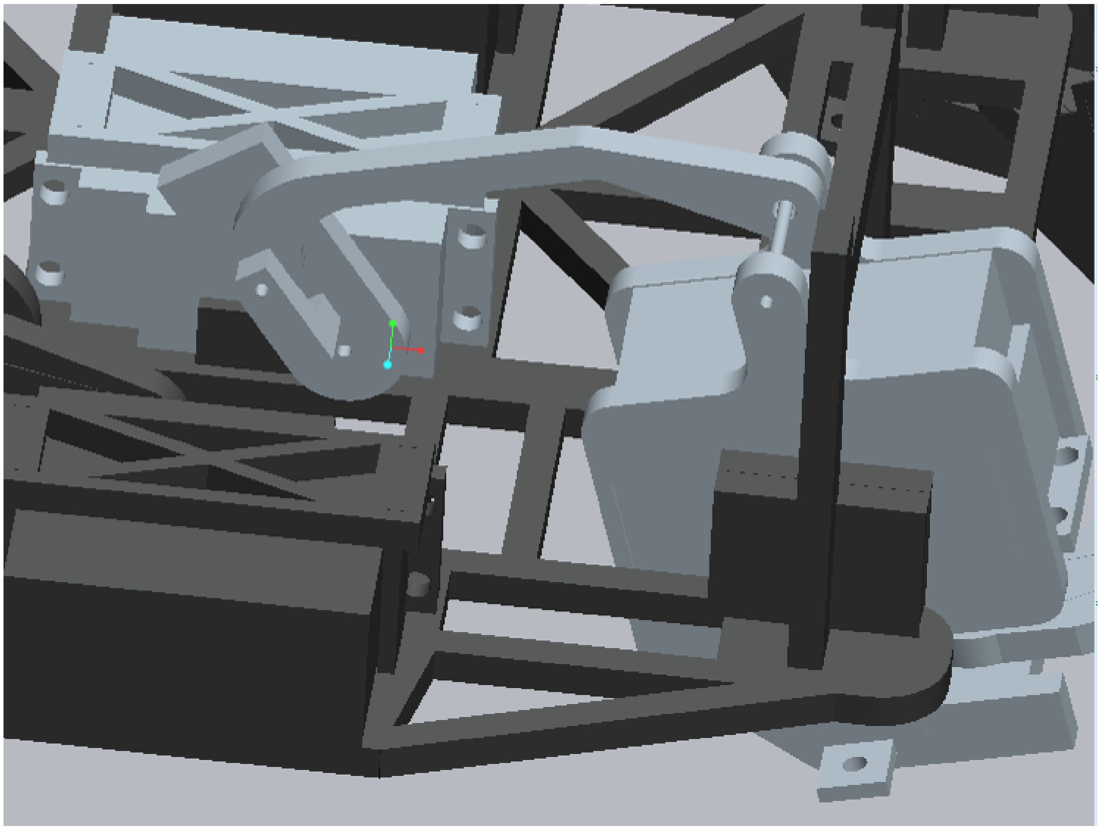


Figure 33 ProE Linkage Model

Since the lift joint of the leg will bear the majority of the weight of the robot, this mechanism greatly increases the payload capacity of the robot, without significantly impacting the operating space of the leg.

Another mechanism was also implemented to augment the lifting power of the lift joint. A system of two extension springs were attached at a radial distance of 2 inches from the center of rotation of the lift joint, and connected to the centerline of the robot. These springs are attached underneath the body of the robot, where the tension of the springs acts to rotate the lift joint such that the leg assembly moves in a downward direction. The spring assembly is shown in Figure 34.

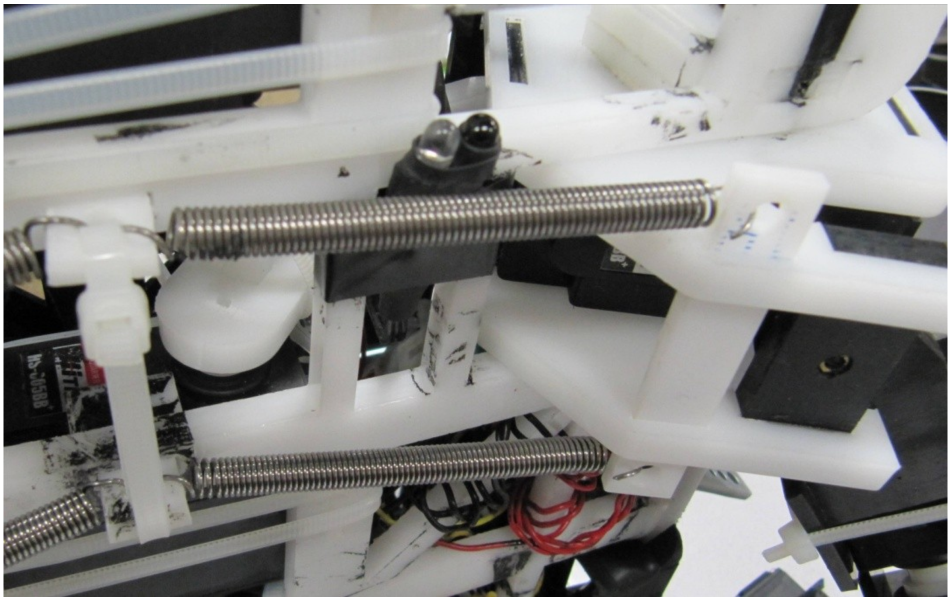


Figure 34 Joint-Assist Springs

These springs are constructed from continuous-length extension springs. They have an outside diameter of 1/4" inch, and an unstretched length of 2.5 inches. The springs are attached such that they are unstretched when the leg of the robot is at its lowest position, and the strings stretch to a length of 4.5 inches when the leg is at its highest position. The springs have a spring constant of 3 lb/in, which means that they are exerting a force of 4.5 lbs each at a distance of 2 inches from the center of rotation of the lift joint. However, with the leg as its highest position, the direction the force exerted by the spring is at a 45 degree angle to the axis of rotation, as shown in Figure 35.

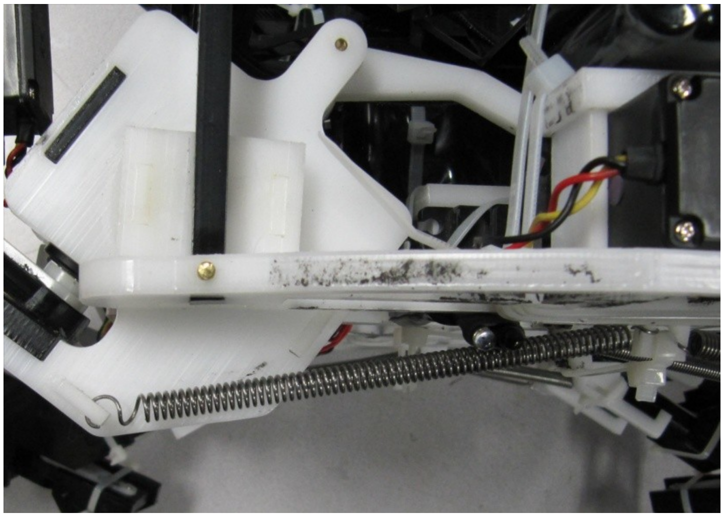


Figure 35 Leg-Assist Springs Fully Extended

This angle results in a final torque on the joint from both springs of 245 oz-in. While these spring result in a large amount of torque necessary to lift each of the robot's legs, it also greatly increases the payload capacity of the robot as a whole.

Because the robot lacks sensors on its feet to detect when the foot is in contact with a surface, some amount of compliance is necessary to ensure that the foot remains in contact with the ground. This compliance was implemented using a spring-driven linear compliant leg, which is shown in Figure 36.

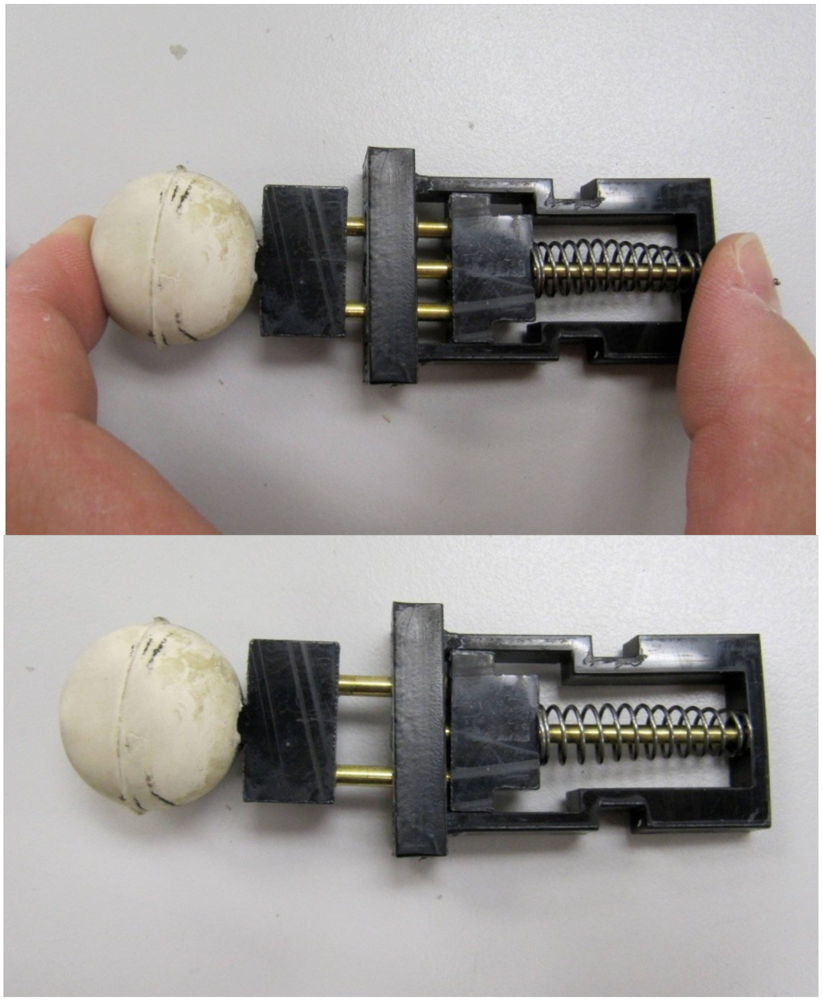


Figure 36 Compliant Lower Leg and Foot Mechanism

This system is constructed of the previously described Delrin, brass rods, and brass tubes, as well as two other components. The first is a linear compression spring, with an unstretched length of one inch and a spring constant of 3.5 lb/in. The mechanism has an available travel space of 0.5 inches, therefore the maximum force exerted by the spring is 1.75 lbs. Due to the small magnitude of this force, even when all four legs of the robot are in contact with the ground and the resultant force on each leg is at its minimum, the leg will be fully compressed. Because this compliant mechanism is only to ensure ground contact between the foot and the ground during motion and no support of the weight of the robot is required, this low spring constant is used.

The second component of the leg mechanism is a simple rubber ball with a hole in the center. These are used as the feet of the robot, due to their high friction, low weight, and round surface. This round surface ensures that the area of the foot in contact with the ground will remain constant independent of the foot or the ground's orientation. These balls are one inch in diameter, and are mounted on 1/2" long 1/4" square pegs manufactured from Delrin as part of the leg-spring mechanism.

3.6 Simulation-Based Sizing of Components

To ensure the robot as designed was capable of the task of stair-climbing, two types of computer simulations were completed. The first was a dynamic simulation of a selected posture the robot will assume when climbing stairs. The second simulation was a Finite Element Analysis of the weakest member of the body as positioned in the dynamically simulated posture, to ensure the member would not fail during normal operation. Both of these simulations were completed utilizing ProE, the FEA analysis completed using the Mechanica application and the dynamic analysis completed utilizing the Mechanism application.

3.6.1 Dynamics Modeling

While a complete dynamic simulation of the robot climbing the stair is possible, this would require extensive computational time, as well as an extensively defined model of the robot itself. Therefore, a dynamic simulation of the posture which requires the most torque from all servos at once was conducted, with the assumption that if the simulation indicated that the robot was capable of the most difficult portion of stair climbing, it would be capable of the entire stair climbing task.

To conduct the simulation, the complete ProE model of the robot was utilized. Each of the components was assigned material properties corresponding to the materials from which they were constructed. The ProE analysis tool was then used to calculate the weight of the model, which was found to be 16.5 lbs. The springs in the compliant legs as well as the leg-assist springs mounted under the body were defined using their physical spring constants, and the feet were designated as having contact with the surface of the stairs. The model is shown in Figure 37 standing in its default position with all constraints defined and shown.

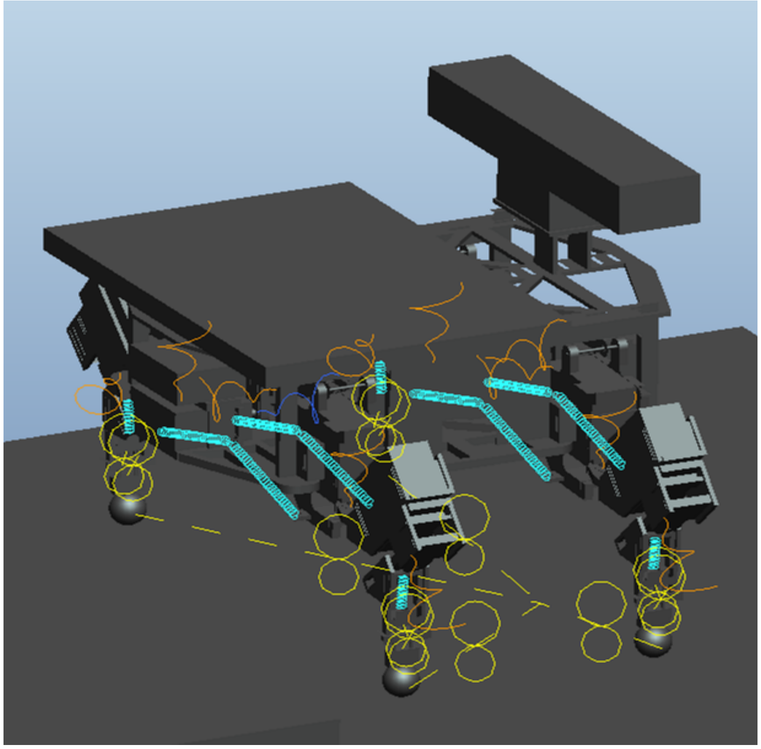


Figure 37 Dynamic Model With Constraints

The most difficult task for the robot when climbing stairs will be to lift itself to a standing position while maintaining balance once it has moved its body completely on to a single step. Because the robot is required to be capable of climbing a step of up to 5 inches in height, this height was used for the simulation, to ensure the robot is capable of lifting its body above a step of this height. The robot's initial position is shown in Figure 38.

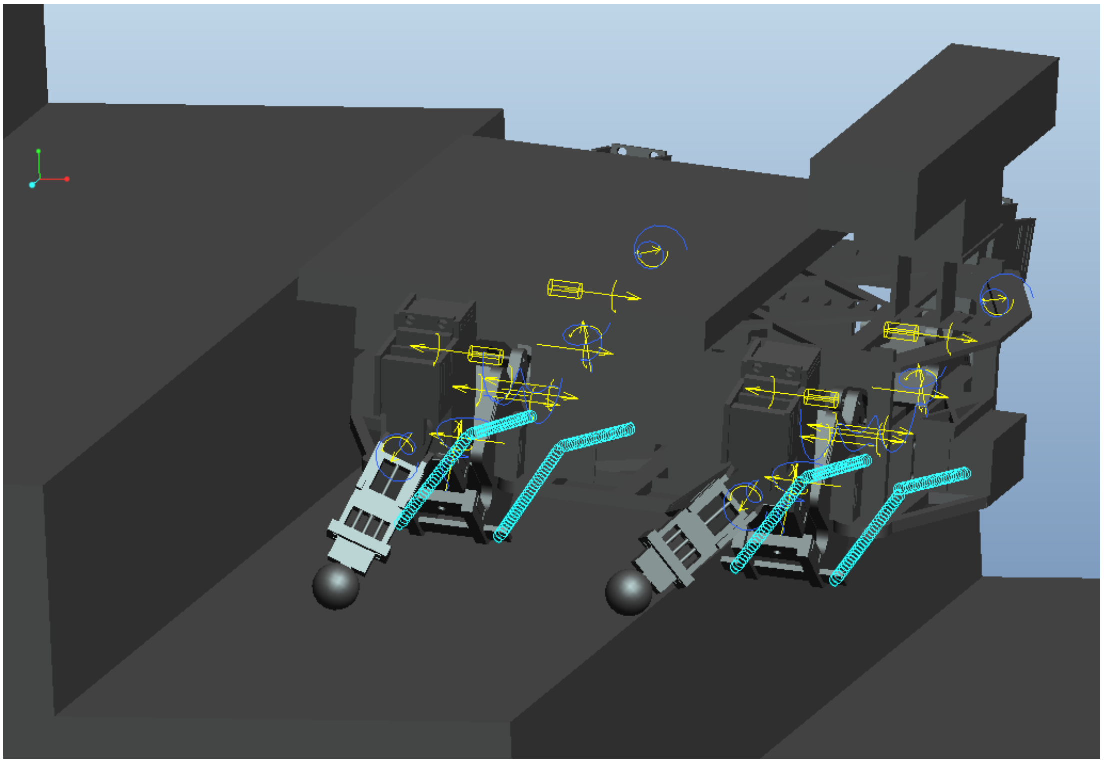
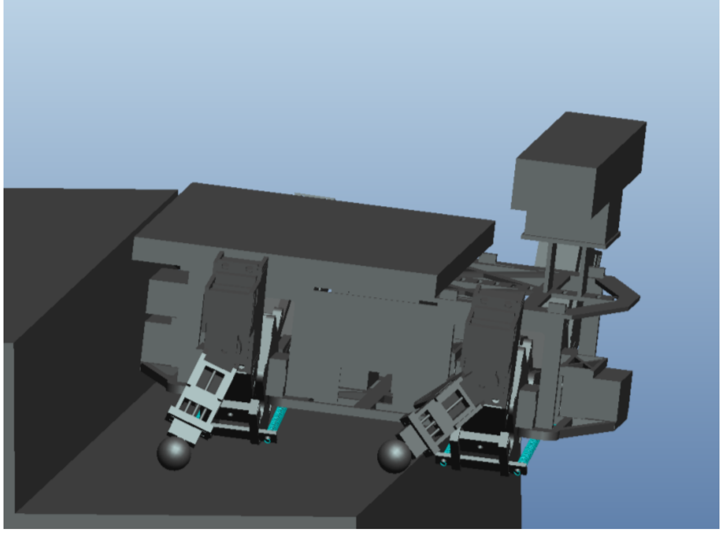
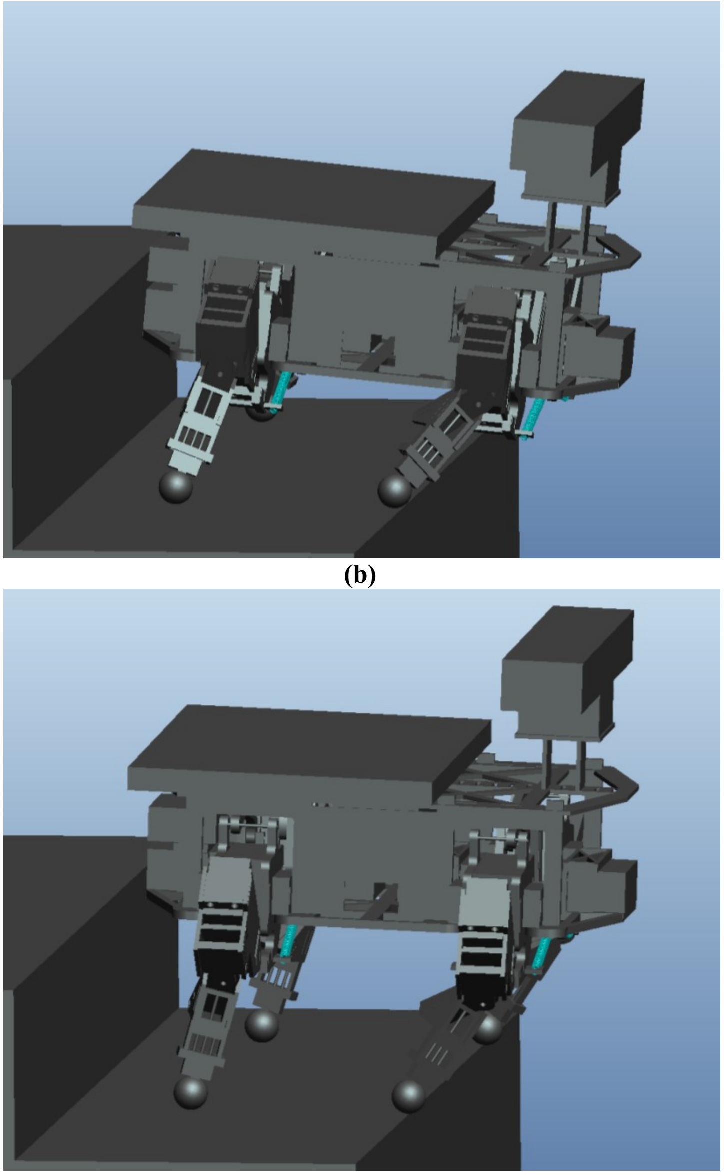


Figure 38 Initial Position for Simulation

In this position the robot's rear feet are in contact with the step closer to the robot's center of gravity than the front feet. This results in a higher load on the rear legs when the robot attempts to stand to obtain its next required position to continue climbing stairs. To determine whether the robot would be capable of completing this motion, each servo was defined as a torque motor capable of the maximum torque of the physical servo. Because the compliant legs are assumed to be compressed for the entire simulation, to reduce the computational time required the legs were constraint at their compressed length. The simulation was set to allow friction between the robot's feet and the stair, with gravity enabled as well. To ensure that the effect of friction would be fully accounted for, a coefficient of friction of 1 was used for both static and kinetic friction. The initial position, the position at an elapsed time of 0.1 seconds, and the position at an elapsed time of 0.2 seconds of the resulting simulation are shown in Figure 39.



(a)



(b)

(c)

Figure 39 ProE Simulation Results: (a) Time=0 sec, (b) Time=0.1 sec, (c) Time=0.2 sec

From the above figure, it can be seen that the robot lifts itself from its lowered position and ends in a standing position, which would allow it to continue to climb the stairs. This indicates that the robot as constructed is capable of the most difficult portion of stair climbing, and therefore is capable of the entire task of stair climbing.

3.6.2 Finite Element-Based Modeling

The component selected to be analyzed using FEA is the upper component of the spring-leg-foot mechanism. This component was selected because by inspection of the robot, it is the component with the smallest cross-sectional area which is placed under the greatest stress during stair-climbing. The ProE model of the component is shown below in Figure 40.

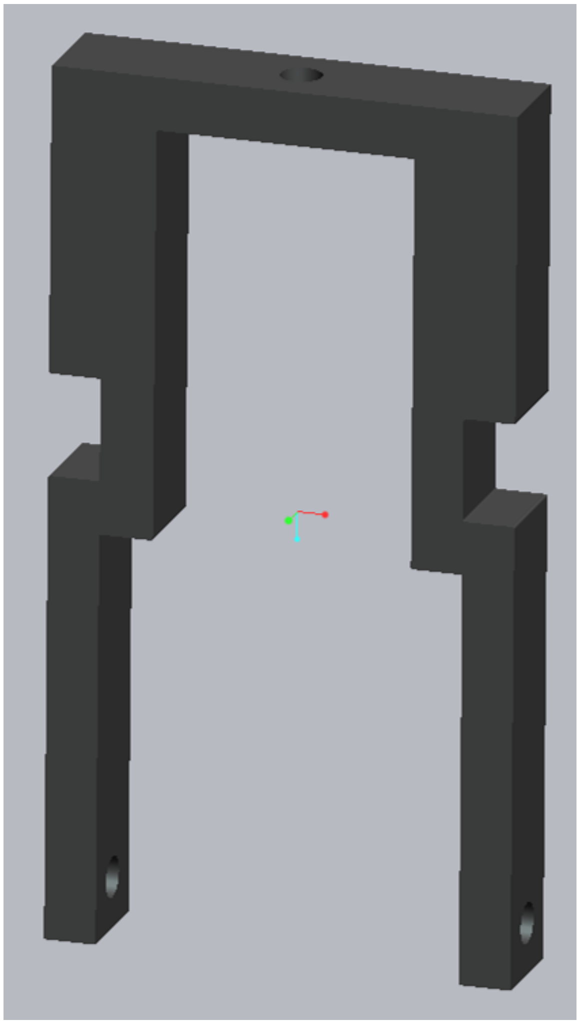


Figure 40 Upper Component of Spring-Leg-Foot Mechanism

When in contact with the ground during normal walking, the load experience by this component is almost completely in the axial direction along both of its leg. However, when the robot is climbing stairs, as shown in Figure 38, the foot contacts the stairs at a close to a right angle to the axial direction of each of the legs, which causes a large amount of shear stress in each leg. To ensure that this component will not fail during normal operation, a simulation will be constructed and the maximum load and safety factor for normal operation will be computed.

For the static load simulation, several assumptions were made. First, it was assumed that while the reaction force is applied to the foot, this is transmitted directly through the foot and brass rods to the lower leg bracket support, where it is directly applied to the component. While the reaction force at a distance from the component, the resultant moment is neglected, since the twisting is negated by the brass rods in contact with the brass tubes. The stress on the lower holes where the component is attached to the lower leg bracket support was also neglected. The component is constrained and loaded as shown in Figure 41.

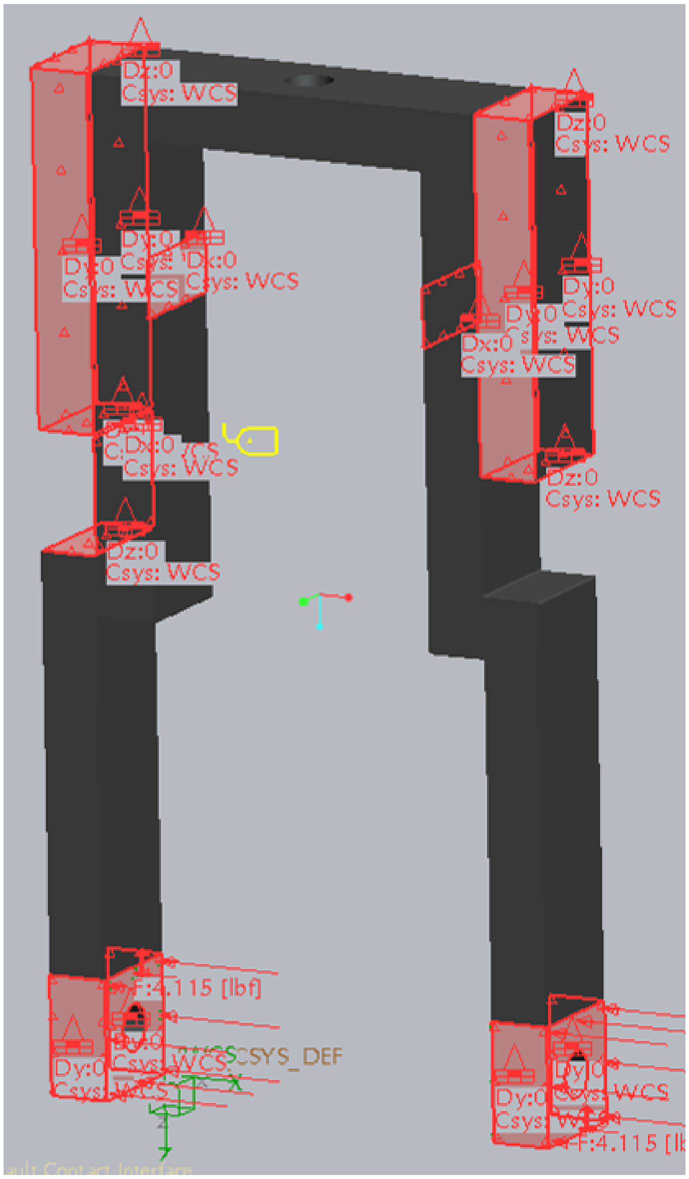


Figure 41 Simulation Constraints

These constraints were chosen based on the physical configuration of the leg. The upper portion of the leg was constrained as it is by the actual geometry, including the small areas constrained by the cable ties which keep the component in place. The load used is a total of 4.125 lbs, distributed over the bottom surface of the component which is in contact with the lower leg bracket support. The load is perpendicular to the face of the surface to which it is applied. While in the dynamic simulation it can be seen that the load would not be completely perpendicular to the leg, however to ensure the leg is designed sufficiently strong for any usage, the perpendicular load was used.

Another constraint applied to the model is a direct link constraint applied to the inner faces of the bottom of each prong of the component. This is to simulate the constant dimension between these two prongs, as the actual component is restricted by the physical constraint of the lower leg bracket support.

The simulation was run utilizing a Multi-Pass Adaptive method, utilizing a maximum polynomial order of 9, and converging on 5% Local Displacement, Local Strain Energy, and Global RMS Stress. The simulation converged on the eighth pass with a polynomial order of 9, and the results are shown in Figure 42.

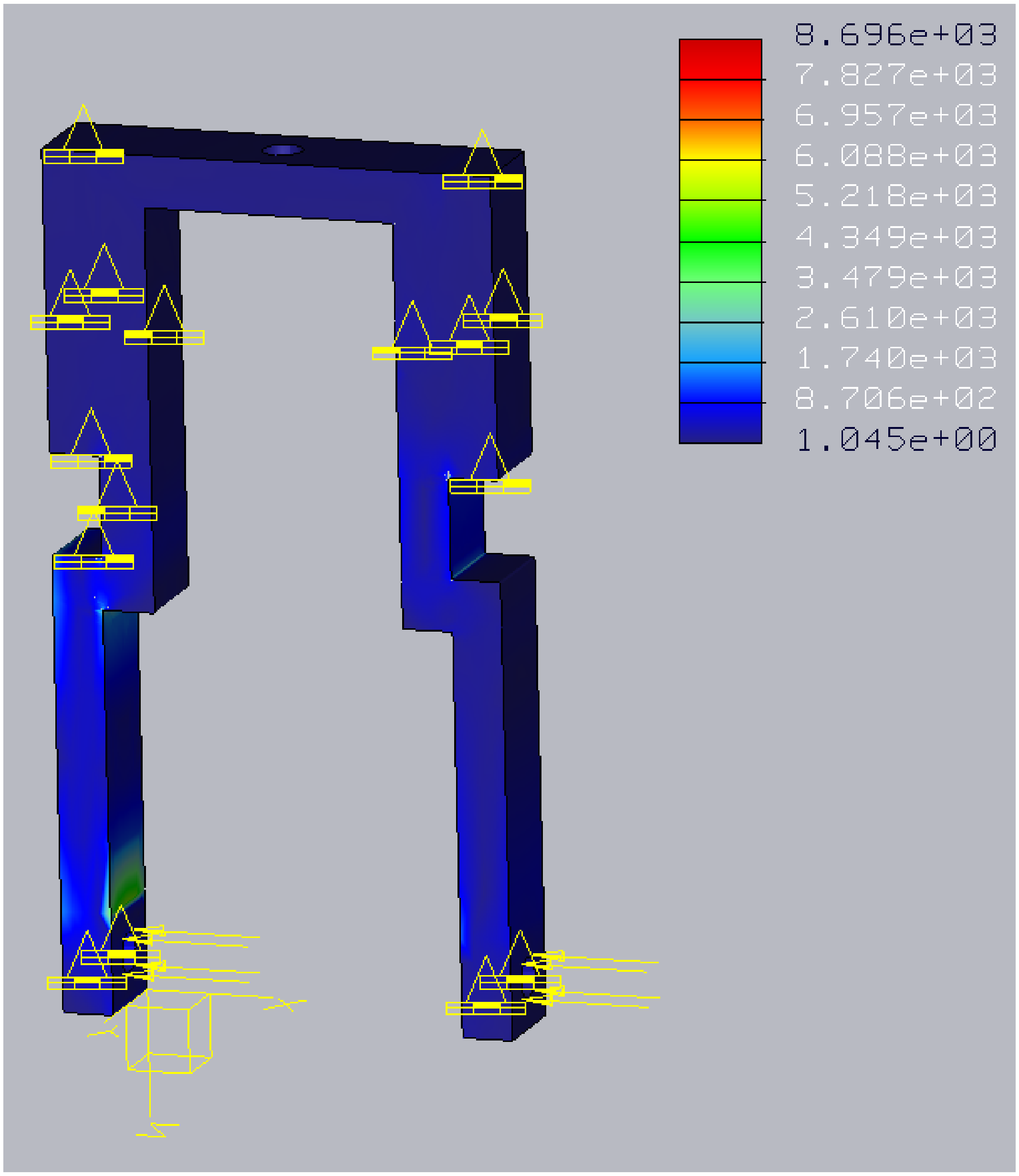


Figure 42 Static Load FEA Results

This simulation shows that when loaded with 4.125 lbs, the highest von Mises stress experienced is expected to be 8,696 psi, which is less than the maximum tensile stress of 10,000 psi for Delrin. This yields a safety factor of 1.15. The exact location of the maximum stress is shown in Figure 43.

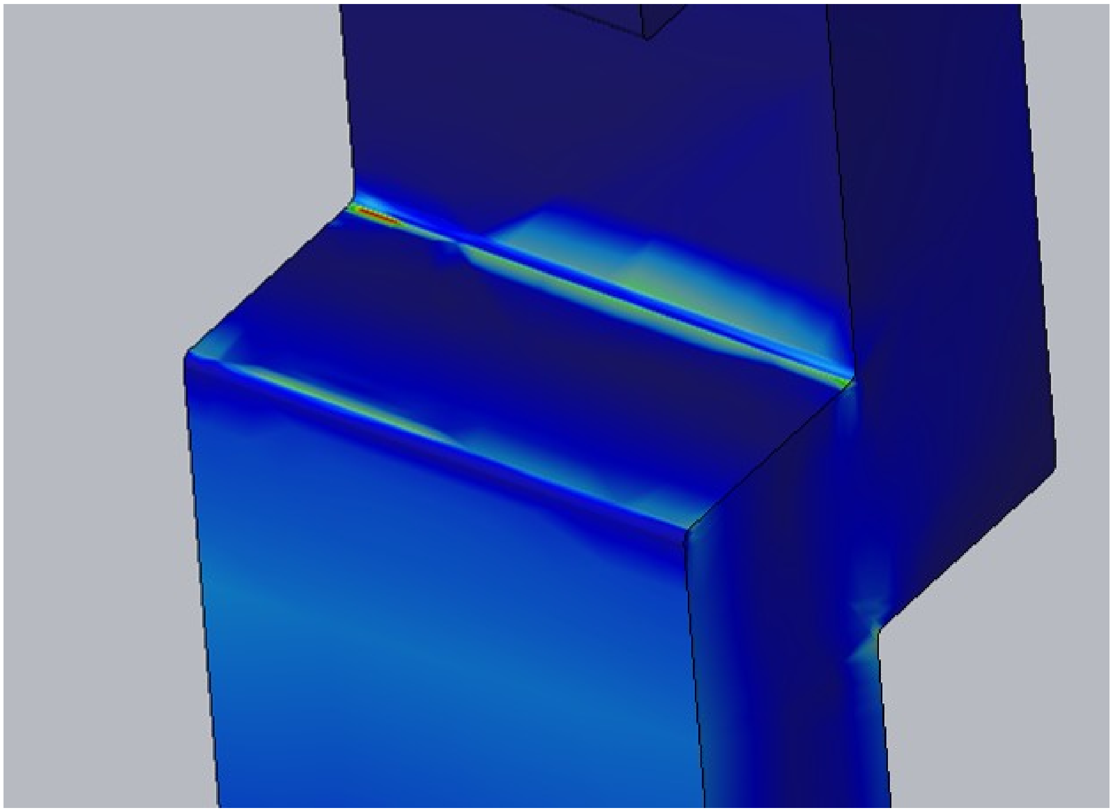


Figure 43 Maximum Stress Location

This shows that the maximum stress location is located at the corner of the center indentation of the left prong of the component.

Because the position and loading of this component which causes this stress occurs only in one posture used by the robot, and in all other postures the load in the direction of this simulation will be less, this simulation shows that the design of the leg is adequate for its use, and failure will not occur during expected operation.

3.7 Summary

The completed robot is detailed below. Figure 44 shows a detailed view of the lift joint linkage system, Figure 45 shows the entire front-right leg assembly, and Figure

46 shows the front and back of the entire assembled robot, including Macbook and Kinect.

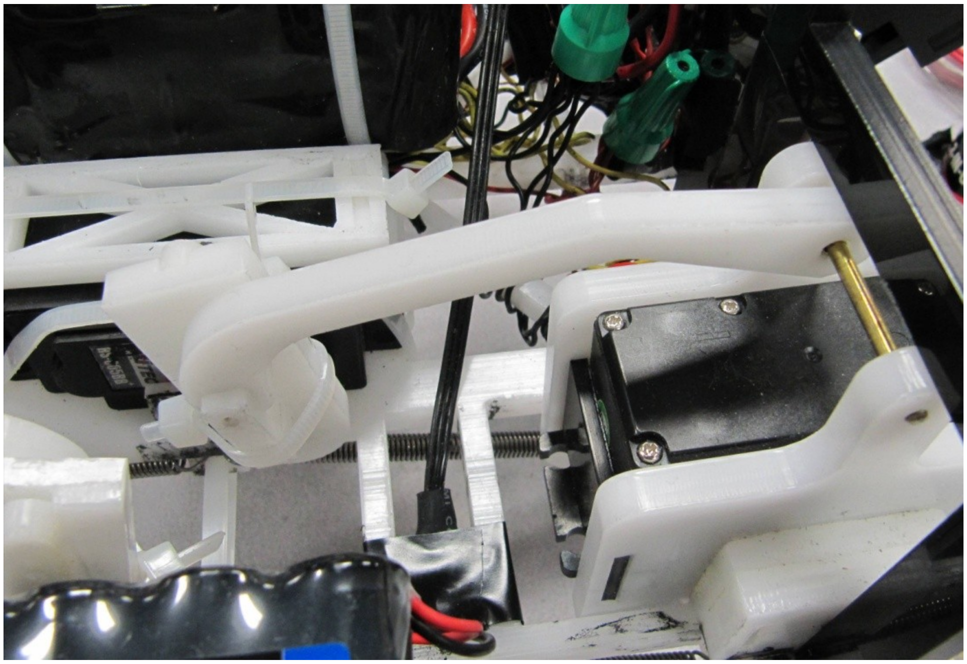


Figure 44 Lift Joint Linkage

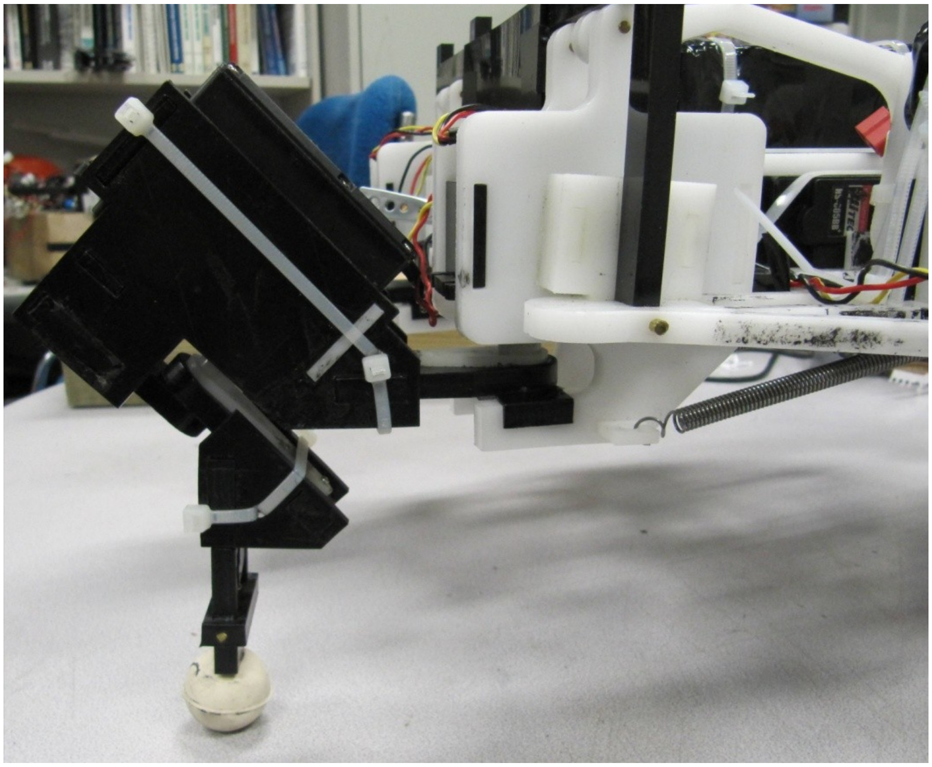
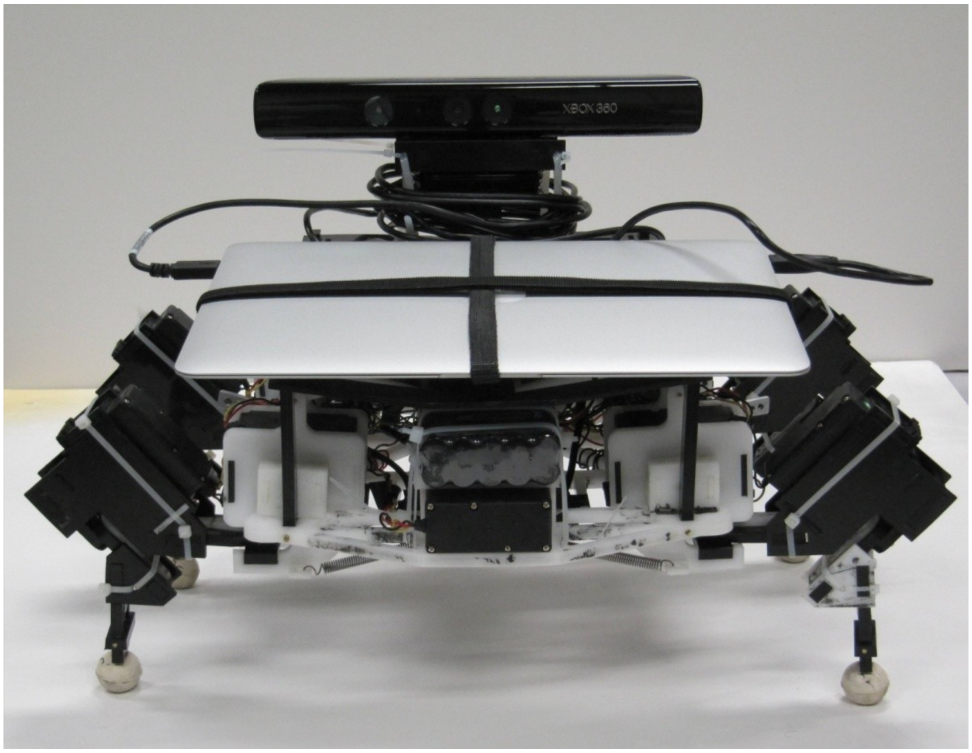


Figure 45 Front-Right Leg Assembly



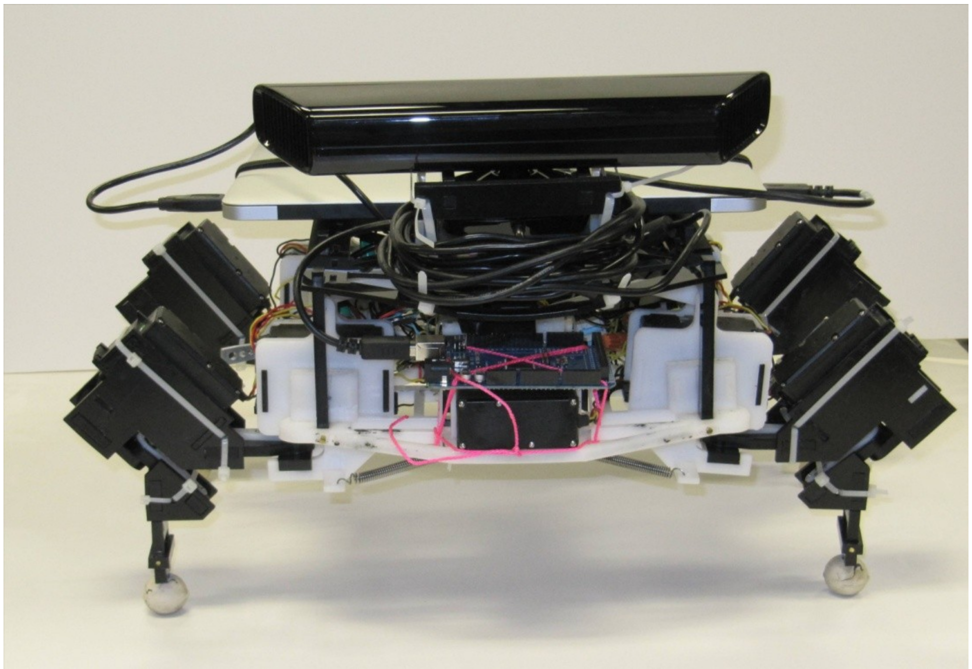


Figure 46 Assembled Robot, Front and Back

The leg lengths used for the robot are 3 inches for each upper leg segment, and each lower leg segment is 4.5 inches uncompressed, 4 inches compressed. The Kinect is mounted on a raised platform behind the Macbook at a height of 1.5 inches. This is to ensure the Kinect's view of the ground in front of the robot is not restricted by the Macbook or the top cover of the robot. The servo power battery packs are mounted in the center of the robot, on top of the center two servos, with a power cut-off switch for each set of servos on each side of the robot body. The Kinect power battery pack is mounted on top to the front servo, and the Arduino controller is mounted on top of the back servo. These components have also been attached to the robot in this manner for optimum weight distribution for stair climbing. While having an overall weight distribution which puts the center of mass forward of the dimensional center of the robot, this distribution assists in stair climbing, as shown in the dynamic model. The final overall weight of the robot is 16.5 lbs, including all batteries, cables, controllers, and sensors, which coincides with the weight generated by the dynamic simulation.

This robot has all required capabilities needed for the task of climbing stairs. It is physically capable of climbing the stairs while carrying all sensors and controllers, while still having the smallest weight possible along with the smallest volume possible. The robot is capable of using the Kinect sensor to both detect and categorize stairs by according to their height, and is therefore capable of climbing stairs of the required heights. This will be demonstrated in the next section.

Chapter 4 – Design and Testing of Parameterized Stair-Climbing Behavior

4.1 Overview

To climb stairs with a quadruped robot, there are multiple approaches, all with advantages and disadvantages. In general, the more in-depth and exact the trajectory planning and foot-placement planning, the more computational power is needed, as well as more complete information about the terrain, robot position, and joint angles is necessary. To reduce the amount of sensor data and computational power needed, the robustness of the behavior must be increased, since without extensive and exact knowledge of the terrain, the behavior used must be able to overcome any obstacles without additional control input or sensor data.

Also, because the goal of this robot is currently only to climb stairs, the required information can be simplified significantly. This is due to the fact that most stairs can be characterized with only two parameters: height and width. While for some staircases the height and width vary from step to step, the majority have constant step height and width. Therefore, it is possible to extrapolate the geometry of an entire staircase from only two parameters.

With this in mind, the robot designed in the previous chapter is designed to climb stairs utilizing a parameterized, static gait. The robot will climb stairs with only the height and width of the stairs provided by manipulating behavior parameters directly related to these two pieces of information, with the parameters modifying previously determined joint angles.

To utilize this parameterized behavior, the robot's sensors must be able to sense the height and width of the stairs, the robot's controller must be able to determine these two parameters from the sensor information and then implement these parameters within the parameterized behavior, which then allows the robot to climb the sensed stair.

In this chapter, a kinematic analysis of the robot is conducted to determine the interactions between the positions of individual feet and the final position of the body of the robot. This information is then used to develop a behavior to climb stairs of a given height and width using a fixed, static gait. This behavior is then parameterized based on the height and width of the stair to be adaptable to any stair.

Once the parameterized behavior is developed, it is implemented on the onboard Arduino microcontroller, where it is programmed to adapt to different stair parameters by receiving simple byte-sized inputs over a serial connection to the Macbook.

The implementation of the Kinect sensor is then discussed, including the utilization of specific Kinect drivers and the MATLAB program. The system is then calibrated to ensure valid measurements of stairs are produced by the Kinect sensor, and the results of the calibration are implemented in the MATLAB program.

The detection and parameterization of stairs is then discussed, followed by the implementation of the entire system. This includes the Kinect sensor detecting the stair, the MATLAB program determines the stair's parameters, which are then transmitted to the Arduino microcontroller, which then controls the robot as it walks forward and climbs the stairs.

4.2 Kinematic Analysis

4.2.1 Leg Analysis

To determine the position of the point where each leg attaches to the body, we must first determine the complete kinematic definition of the leg. Since each leg has 3 active and one passive degree of freedom, the position of each leg's attachment point will be determined by 4 variables. These will determine the kinematic definition of the kinematic definition of the leg.

To perform the kinematic analysis of the leg, reference coordinate systems must be determined. These are shown below in Figure 47.

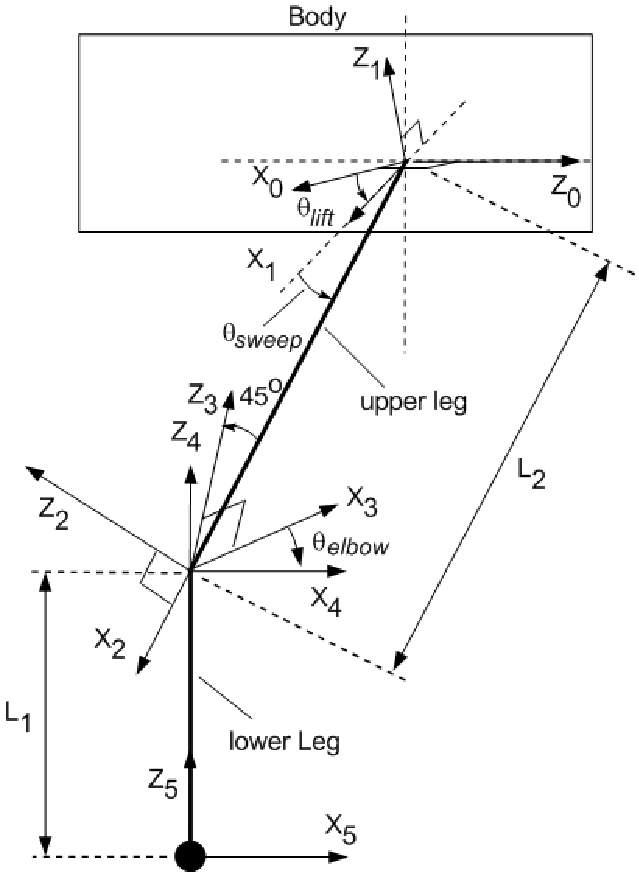


Figure 47 Leg Reference Coordinate Systems

With these coordinate systems it is possible to derive the equation expressing the position of the foot relative to the leg's attachment point as a function of the leg angles and the length of the compliant leg using the Denavit Hartenberg (DH) Convention.

The DH Convention is used for selecting reference frames when evaluating joint-linkage combinations, and is commonly used in robotics, particularly for robotic arm and leg evaluation. The DH Convention uses a set of four transformations to represent each homogeneous transformation, with each transformation represented by a DH parameter. These four parameters are:

- d: offset along the joint axis from the reference joint to the common normal
- θ: angle about the joint axis from the reference joint to the next joint axis
- a: length along the common normal between the two joints
- α: angle rotated about the common normal, from the reference joint to the next joint axes

where the common normal is the vector between the two joints, perpendicular to both joint axes.

Using DH parameters, the position of each coordinate of the leg's separate coordinate systems is defined relative to the coordinate system it connects to, in order from the body to the foot. This coordinate transformation is performed using the transformation matrix defined in (1).

$$[T] = \begin{bmatrix} & [R] & & \vec{T} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

In the above equation, $[R]$ is the 3x3 rotational transformation matrix and $\vec{T}$ is the 3x1 translation vector. Using this matrix, a vector in a defined coordinate system can be expressed in terms of the unit vectors of another defined coordinate system if the rotation and translation of the new coordinate system relative to the original coordinate system is know. This transformation is performed using (2).

$$\vec{P}_{n+1} = [T] \cdot \vec{P}_n \tag{2}$$

where $\vec{P}_n$ is the positional vector in the n coordinate system, and $\vec{P}_{n+1}$is the positional vector of the same point in the n+1 coordinate system.

These equations are now used to define the position of the robot's foot relative to the attachment point on the robot's body. To transform from the body's coordinate system a coordinate system attached to the lift rotational joint, the DH Parameters shown in Table 6 will be used.

Table 6 Leg DH Parameters

Coordinate System	i	θ_i	d_i	α_i	a_i
Lift	1	θ_{lift}	0	$\pi/2$	0
Sweep	2	θ_{sweep}	0	$\pi/2$	L_2
Plane Adjust	3	$-3\pi/4$	0	$\pi/2$	0
Elbow	4	θ_{elbow}	0	$\pi/4$	0
Foot	5	0	$-L_1$	0	0

It should be noted that for these calculations the small distance between the lift and sweep axes was neglected for simplicity.

Using these parameters, the transformations between each coordinate system can be developed. The transformation from the body coordinate system to the lift coordinate system is shown in (3).

$$\vec{P}_1 = \begin{bmatrix} \cos\theta_{lift} & 0 & \sin\theta_{lift} & 0 \\ \sin\theta_{lift} & 0 & -\cos\theta_{lift} & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{P}_0 \tag{3}$$

where

$$\vec{P}_1 = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}, \vec{P}_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

The transformation from the lift coordinate system to the sweep coordinate system is shown in (4).

$$\vec{P}_2 = \begin{bmatrix} \cos\theta_{sweep} & 0 & \sin\theta_{sweep} & L_2\cos\theta_{sweep} \\ \sin\theta_{sweep} & 0 & -\cos\theta_{sweep} & L_2\sin\theta_{sweep} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{P}_1 \qquad (4)$$

The transformation from the sweep coordinate system is to the plane adjust coordinate system is shown in (5). This transformation is necessary to align the sweep coordinate system with the elbow coordinate system, due to the 45 degree angle the elbow rotation axis makes with the upper leg length.

$$\vec{P}_3 = \begin{bmatrix} -\cos\left(\frac{3\pi}{4}\right) & 0 & -\sin\left(\frac{3\pi}{4}\right) & 0 \\ -\sin\left(\frac{3\pi}{4}\right) & 0 & \cos\left(\frac{3\pi}{4}\right) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{P}_2 \qquad (5)$$

Now that the coordinate system is aligned with the elbow rotational axis, the transformation to the elbow coordinate system is given by (6). This includes a rotation to align with the lower leg.

$$\vec{P}_4 = \begin{bmatrix} \cos\theta_{elbow} & -\cos\left(\frac{\pi}{4}\right)\sin\theta_{elbow} & \sin\theta_{elbow}\sin\left(\frac{\pi}{4}\right) & 0 \\ \sin\theta_{elbow} & \cos\theta_{elbow}\cos\left(\frac{\pi}{4}\right) & -\cos\theta_{elbow}\sin\left(\frac{\pi}{4}\right) & 0 \\ 0 & \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{P}_3 \tag{6}$$

To translate the coordinate system from the joint to the foot, the foot adjust transformation is used, including the translational component due to the length of the lower leg. This transformation is shown in (7).

$$\vec{P}_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{P}_4 \tag{7}$$

This gives the position of the foot relative to the position of the leg attachment point on the body.

These transformations together transform the input of the joint angles and leg length into a single vector from the leg attachment point on the body to the position of the foot relative to the body. The full transformation is given in (8).

$$\vec{P}_5 = [T_0^5] \cdot \vec{P}_0 \tag{8}$$

where $[T_0^5]$ is the transformation matrix constructed from all five transformations required, shown in (9).

$$T_0^5 = T_4^5 \cdot T_3^4 \cdot T_2^3 \cdot T_1^2 \cdot T_0^1 \quad (9)$$

Having determined the position of each foot relative to the attachment point at the body, the kinematic model for the entire robot can be developed.

4.2.2 Full body Analysis

Utilizing the kinematic analysis of each leg, it is possible to determine the position of the center of mass of the robot using the positions of three out of four leg attachment points on the body. The body can be defined using the position of three joint attachment points, which can in turn be defined by foot position and the three controllable joint angles of each leg. The three main degrees of freedom that will be derived are the body height and the rotation of the body around two perpendicular axes. These are shown below in Figure 48.

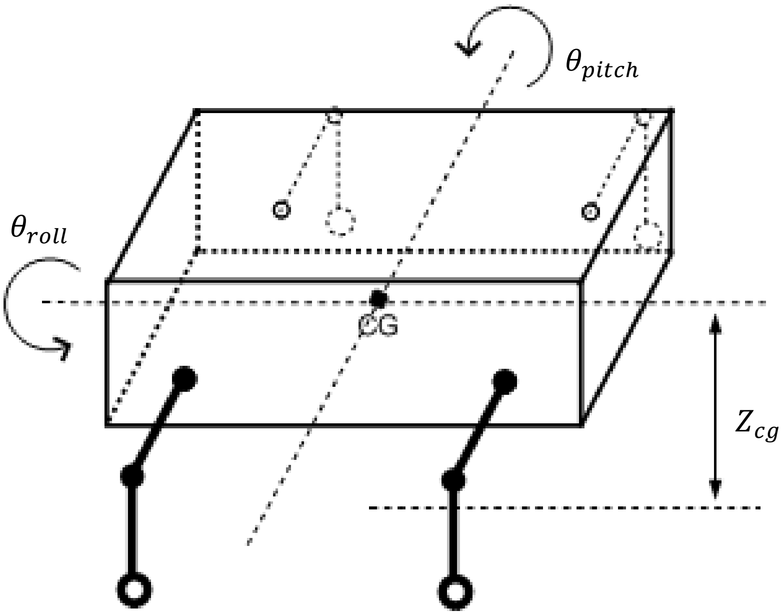


Figure 48 Body Dynamics

To determine the three variables from the above figure, we will assume a small angle approximation. This is a valid assumption, since the robot body should not roll or pitch a large amount due to the method of stair-climbing selected. If the intended stair-climbing behavior included the robot body spanning more than two stairs at a time, than the small angle approximation would not be appropriate, since the body of the robot would rotate approximately to the angle of the staircase, which in most cases is thirty to forty degrees. However, since the intended behavior will keep the robot's body approximately parallel with the surface of the stair, the small angle assumption is valid and can be utilized.

While the equation to determine the position of each leg attachment point has been derived, only three of these are needed to determine the position of the body.

Since the center of gravity is centered on the body, the height of the center of gravity can be found by adding the average of the heights of the two leg attachment points on each side of the robot, as shown in (10).

$$Z_{cg} = \frac{X_2 - X_1}{2} + \frac{X_3 - X_1}{2} + X_1 \tag{10}$$

where X_1 is the leg attachment point on the side of the robot with both attachment points being utilized which is also parallel to the attachment point being utilized on the opposite side of the robot, and X_2 and X_3 are the other two attachment points in use.

To find the roll angle, the height of the two leg attachments which are opposite each other on either side of the robot are used, as shown in (11).

$$\theta_{roll} = \sin^{-1}\left(\frac{X_{left} - X_{right}}{L_{width}}\right) \tag{11}$$

where X_{left} and X_{right} are the heights of the left and right sides, and L_{width} is the width of the robot, measured between the leg attachment points. As previously discussed, this assumes a small angle of rotation θ_{roll}. This equation also keeps with the sign convention established in Figure 48, right rolls to the right being positive angles while rolls to the left are negative angles.

For the pitch angle, the height of the two leg attachments which are on the same side of the robot are used, as shown in (12).

$$\theta_{pitch} = \sin^{-1}\left(\frac{X_{front} - X_{rear}}{L_{length}}\right) \tag{12}$$

where X_{front} and X_{rear} are the heights of the front and rear leg attachment points of the robot, and L_{length} is the length of the robot, measured between the leg attachment points. This equation also keeps with the sign convention of Figure 48, where the backward pitch of the robot is positive, and the forward pitch is negative.

With these equations now defined, it is possible to describe the body height and orientation using the joint positions and foot locations for three of the robot's legs.

4.3 Behavior Concept

As previously discussed, the robot will utilize a static stair-climbing behavior which maintains a stable body position. A static gait indicates that the robot always has at least three points of contact with the stair at any time. This enables the robot to maintain its stability when climbing by simplifying the calculations and control mechanisms required.

4.3.1 Overall Behavior Design

The stair-climbing behavior developed consists of three distinct phases of movement, with each phase repeated in sequence for each step. Because the robot's final position at the completion of phase 3 is also the starting position for the beginning of phase 1, these steps can be repeated for any number of steps without intermediate control or sensing.

4.3.2 Phase One

To begin climbing stairs, the robot stands in front of the step with all four feet in contact with the ground and each leg at its zero position. The zero position for each leg is defined as the upper leg segment being parallel to the ground, and the lower leg segment is perpendicular to the ground, with the tow leg segments also perpendicular to each other. When all four legs maintain this position, the pitch and roll angle of the body are both zero. This initial position is shown in Figure 49.

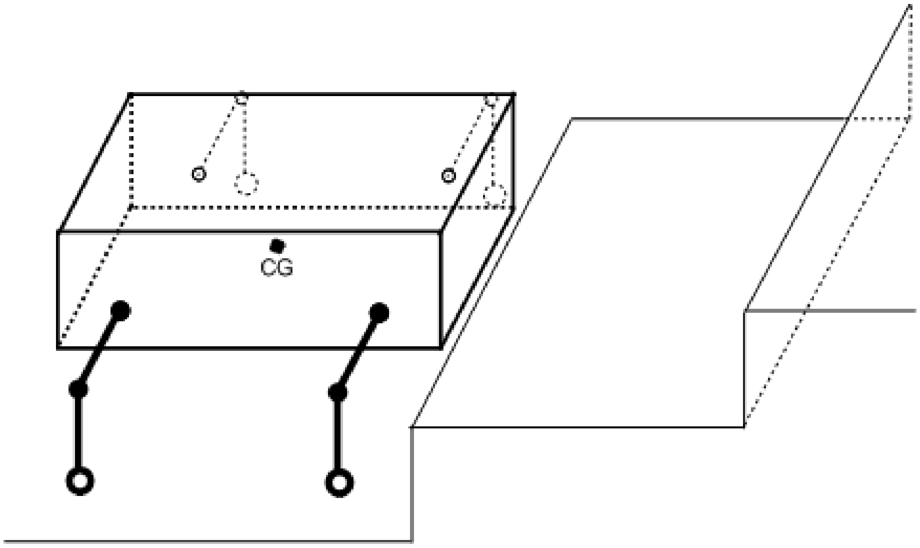


Figure 49 Robot Initial Position

To begin climbing, the robot first rotates the lift joints of all legs downward, sliding the feet inward toward the body while lifting the body vertically. In this movement, the front legs lift slightly more than the rear legs, which slightly inclines the body backwards. This serves to increase the height of the front leg attachment points as well as lift the center of gravity of the body above the stair. This position is shown in Figure 50.

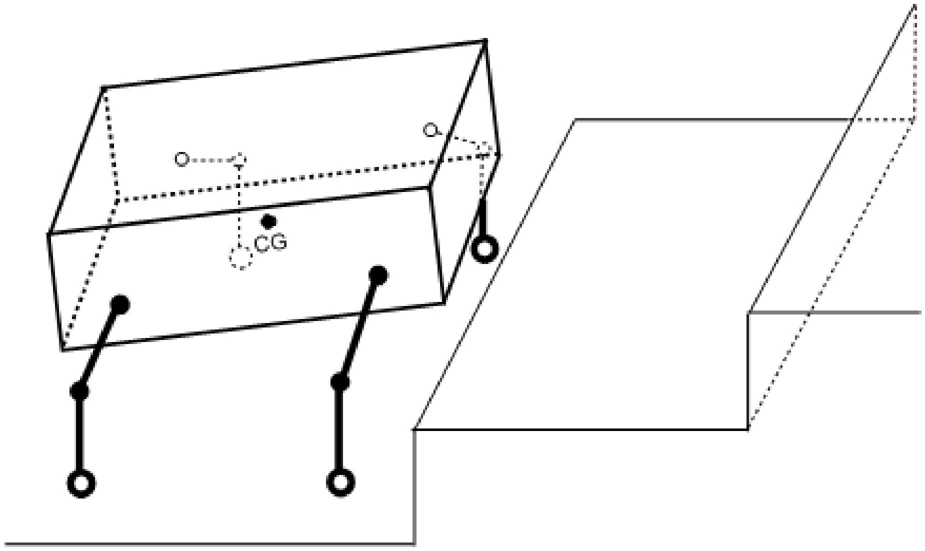


Figure 50 Body Lift Position

Next, the front left leg is lifted and the foot placed on the top surface of the step. This is accomplished in three distinct movements. First, the lift joint is rotated in the upward direction, while the elbow joint is rotated forward and the sweep joint is rotated slightly toward the rear of the body. This slight backward motion is to ensure the foot, which is moved toward the front of the body through the rotation of the elbow, does not come into contact with the front face of the step. Once this motion has occurred, the sweep joint rotates toward the front of the body, placing the foot over the top of the stair. The elbow joint is then rotated slightly downward while the lift joint rotates downward, placing the foot in contact with the top of the stair and allowing the leg to support the body. Once this set of motions is completed, it is repeated by the front right leg. In the resulting position, both front feet are resting on top of the stair, and the body is raised and pitched slightly toward the rear. This position is shown in Figure 51.

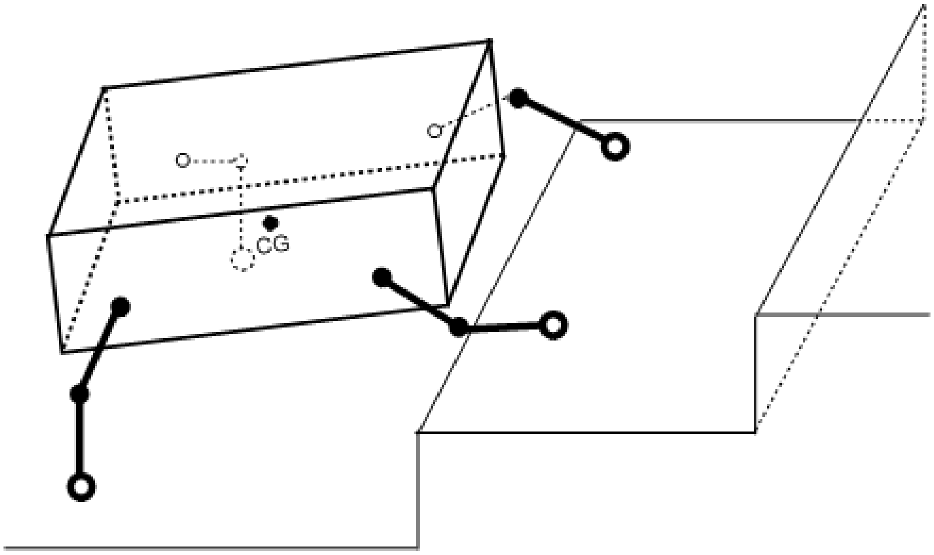


Figure 51 Front Legs Up Position

This position completes phase one. The initial position for subsequent steps is varied slightly from the first step, and will be discussed at the end of Phase 4.

4.3.3 Phase Two

This phase consists of several similar movements which traverse the robot's body forward, ending with the robot in the same position as in the beginning of the phase, but with the center of gravity over the step and the rear legs touching the step.

To move forward, the robot begins by rotating all sweep and elbow joints backward. This maintains the current foot position, but moves the body's center of gravity forward while remaining statically stable. This movement results in the position shown in Figure 52.

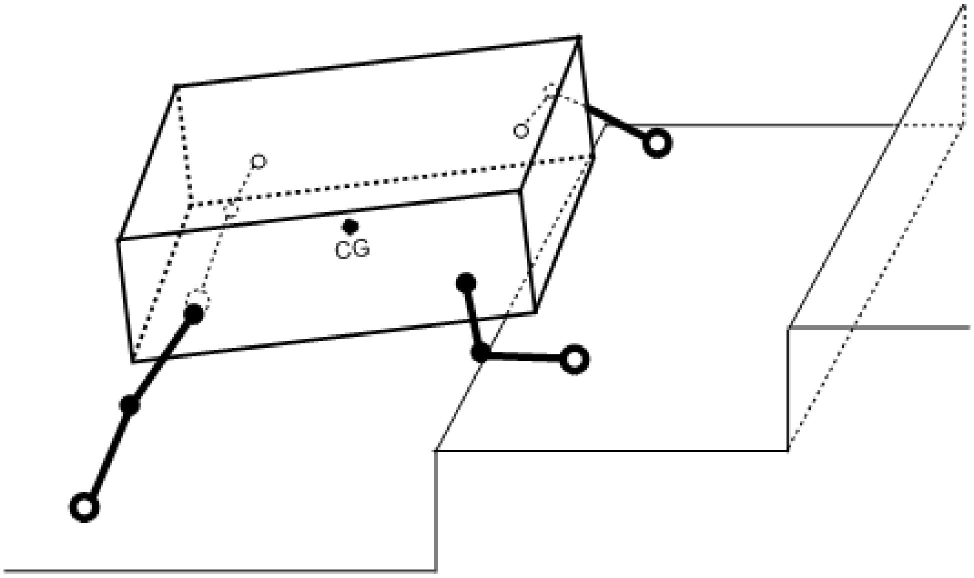


Figure 52 Body Forward Position

With the body now moved forward, each individual leg is lifted and moved forward. This is accomplished by initially maintaining the elbow and sweep angles of the leg while rotating the lift joint upwards, then moving the foot forward and placing it back on the ground or stair by rotating the lift joint. The elbow and sweep angles are initially maintained constant to decrease the friction which would be caused by moving the foot forward at the same time as it is lifted, or “dragging”. The front left leg is the first moved, followed by the right rear, front right, and left rear, respectively. Once all four legs have been moved forward, the robot is again in the Front Legs Up position, but with the body of the robot moved further onto the step. This position is shown in Figure 53.

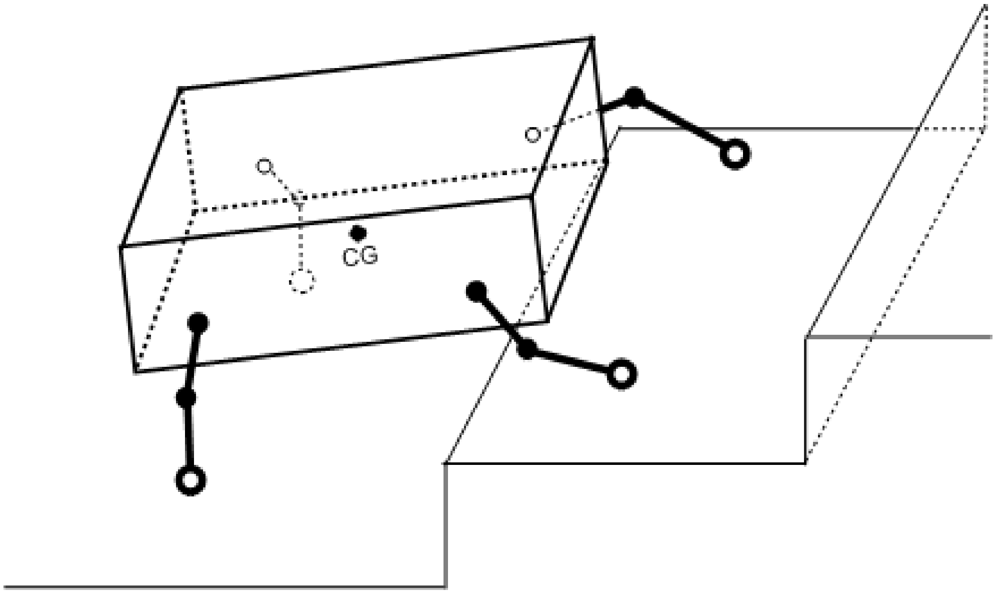


Figure 53 Legs Forward Position

This sequence of movements is repeated two more times, with the end result being the robot's body positioned as far forward on the step as possible in the Body Forward Position. However, the second of these sequences is varied, in that the order of the movement of the legs is changed by reversing the side, such that the front right leg moves first, followed by the left rear, front left, and right rear. This ensures that any variation in the surface roughness of the stairs or ground does not impact the final position of the robot at the end of Phase Two.

4.3.4 Phase Three

This phase begins with the robot in the Body Forward position, with both rear legs in contact with the side of the stair. To position all four feet on top of the stair, both rear legs must be raised and placed on the top of the stair. To accomplish this, the rear of the robot is lowered by simultaneously raising both rear legs. The underside of the body immediately in front of the rear leg joint attachment positions rests on the edge of the stair itself, which serves to maintain static stability of the robot, despite only the front two feet being in contact with the stair. The rear feet are then raised above the step using movements identical to the movement of the front two legs during the Front Legs Up movement of Phase One. The rear feet are then placed on the step by rotating the elbow joints downward. This position is shown in Figure 54.

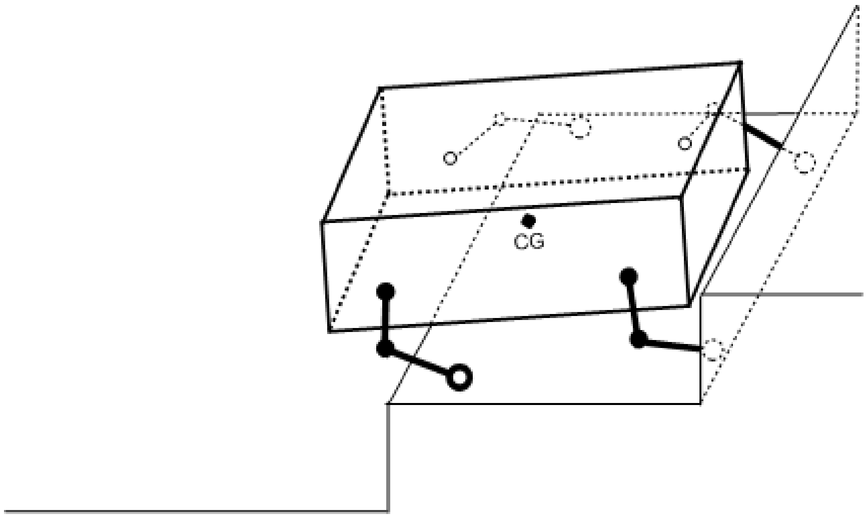


Figure 54 Rear Legs Up Position

Once all four feet are on the same stair, the robot lifts its body by rotating both the elbow and lift joints downward, as well as rotating the sweep joints slightly backward, which brings the robot to a position very similar to the initial Body Lift position of Phase One. The position in Phase Three varies slightly in that the rear feet are placed slightly forward of the Phase One position, due to the limited physical space available on one on the step. The final position is shown in Figure 55.

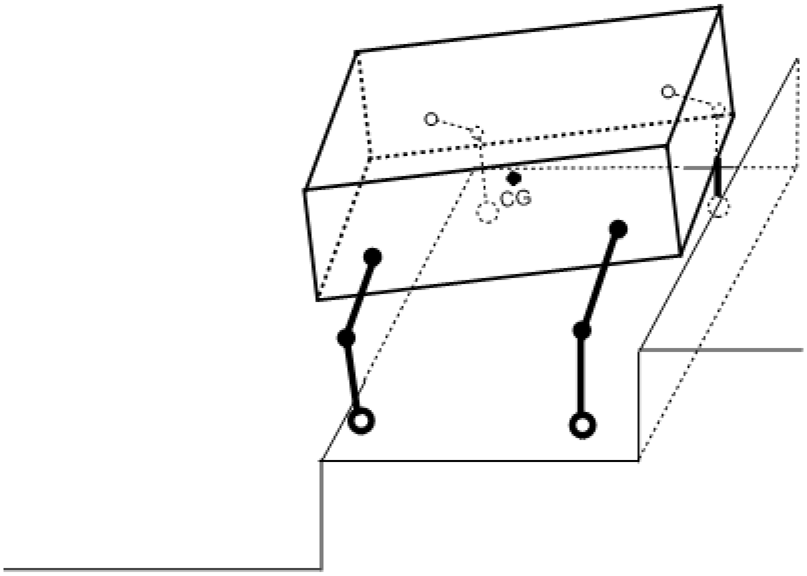


Figure 55 Standing Position

The robot is now in the Body Lift position of Phase One, and continues to repeat the stair-climbing behavior until the robot reaches the top step.

4.4 Behavior Parameterization

The behavior developed is fairly robust, in that while it is designed for a step height of five inches, it is capable of climbing stairs of approximately one half of an inch both taller and shorter than that height. However, it is not effective at climbing stairs with a height of less than 4.5 inches, therefore the behavior is parameterized to climb any sized stair.

4.4.1 Stair Height Parameterization

To parameterize the stair-climbing behavior, which was developed for a stair height of five inches, a system of parameters for each joint was developed which modifies the joint angles which are affected by the change in stair height. Because the lift angle is most affected by the change in stair height, parameters were only applied to those joint angles, with the sweep and elbow joint angles remaining constant. Also, the parameterization was only applied to the lift angles where the foot is in contact with the ground or step. This ensures that when the leg and foot are lifted, unwanted contact with the stairs or ground is avoided. The parameterization factor is shown in (13).

$$\theta_{lift_p} = \theta_{lift} \cdot \left(\frac{h_{stair}}{4.5}\right) \tag{13}$$

where θ_{lift} is the lift angle from the previously defined stair-climbing behavior designed for a stair height of 5 inches, and h_{stair} is the height of the stair detected by sensors. The height of the stair being climbed divided by 4.5 is the parameter which determines the change to the behavior, and its purpose is to scale the height of each foot's placement to coincide with the height of the stair. This effectively maintains the body height and pitch angle, regardless of the height of stair being climbed. The original behavior was developed for a stair height of 5 inches, therefore the factor of 4.5 was used to provide for a margin of error, in that using a smaller factor will result in the angles used being larger than needed, but still able to successfully climb the stair. For instance, if the stair height is 5 inches, but the sensors detect a height of 4.5 inches, the parameterized behavior will lift the legs using angles that will result in a successful climb of the 5 inch tall stair. A graph of the lift angles for the left rear leg is shown to illustrate the parameterization in Figure 56.

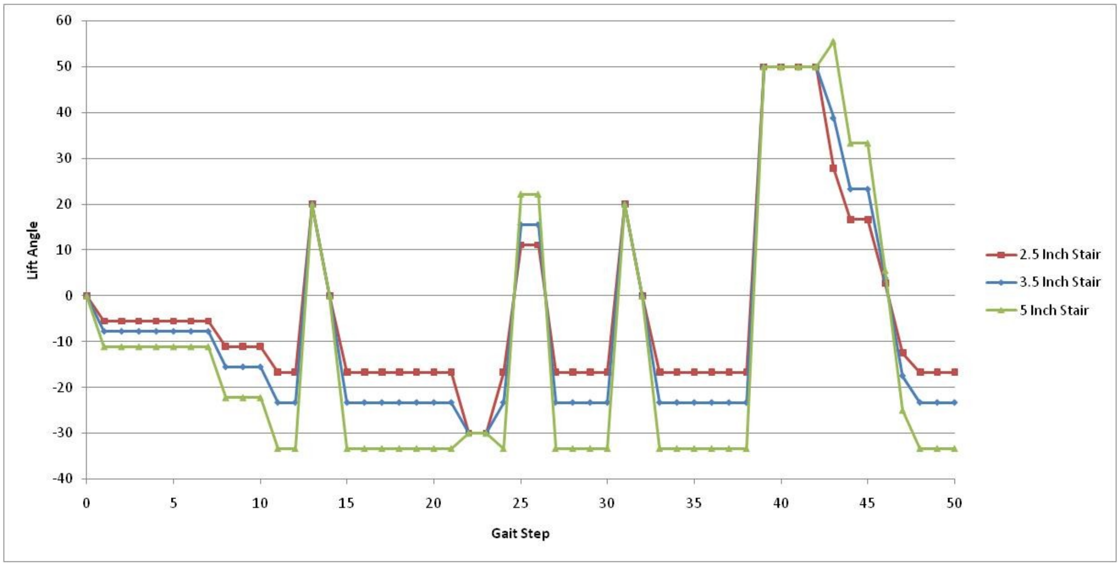


Figure 56 Parameterized Left Rear Lift Joint Angles

In the above figure, the lift angle is shown after parameterization for three different stair heights. The lift angle is parameterized using (13) for all steps, with the exception of four positions. The first instance where the parameterization factor is not applied is at steps 12 and 13, where the left rear leg is being lifted to take a step forward during Phase Two of the behavior. The second and third instances where the factor is not applied are also during Phase Two, while the leg is being lifted to take a step forward. The fourth instance is during Phase Three, when the rear legs are being raised simultaneously to be placed on the step.

4.4.2 Stair Width Parameterization

The behavior was also adapted to account for variation in stair widths, to ensure that the behavior can accommodate any stair width. Since the required motions for climbing a single stair are not affected by the stair width, the stair width parameterization does not affect the original behavior or the height sensitive parameterized behavior.

The width parameterization was accomplished by inserting a simple forward walking gait with a very small step length into the overall behavior between the end of Phase Three and the beginning of Phase One. The simple distance sensors mounted on the front of the robot are capable of detecting when the robot is immediately in front of a step, therefore after every step the robot checks to see if another step is detected. If a step is detected, then the robot utilizes the forward gait with small step length to move forward until the next step is detected. Once the step is detected by both the left and right sensors, the parameterized stair behavior is begun again, using the same stair height parameterization used for the previously climbed step.

The last step of the parameterization is a limit placed on the distance which the robot will move forward using the slow forward gait. If the front sensors do not detect another step within three iterations of the forward gait, during which the robot covers approximately 4 inches, then the robot ceases the overall stair-climbing behavior, and for the purposes of this study the robot is set to assume its zero position and standby for further instruction of the laptop controller.

With the parameterization as delineated in this section, the robot is capable climbing any stair with a detected height of 2.5 to 5 inches, and any width under 14 inches, though the 14 inches can be expanded as needed, since it is not dependent on the robot's physical characteristics.

4.5 Behavior Execution

To execute the parameterized behavior, the stair parameters must be estimated by the MATLAB control program utilizing data from the Kinect sensor. Once the parameters are obtained, this information must be incorporated into the behavior, which then must be implemented utilizing the Arduino microcontroller. This process is now described.

4.5.1 Stair Parameter Estimation

As previously discuss, the Kinect sensor is used to detect the height and width of the stairs using its infrared depth sensor. This is accomplished by reading the depth information stream from the Kinect, using an image analysis algorithm to analyze a single image frame from the depth steam to determine the height and width of the stair, then transmitting this information to the Arduino microcontroller.

To read the information stream from the Kinect, the Microsoft Kinect Software Development Kit (SDK) Beta 2 version 1.0 software was installed on the Macbook laptop. To perform the image analysis of the depth image, MATLAB software was utilized, therefore a method to import the Kinect data stream into MATLAB was needed. To accomplish this, open-source files developed and uploaded to the MATLAB file exchange website by Dirk-Jan Kroon were utilized [73]. These files permit MATLAB to read and store the depth images generated by the Kinect at up to 30 frames-per-second in real time.

The Kinect generated depth images are grayscale, with darker shades of gray representing surfaces which are closer to the Kinect, with lighter shades of gray corresponding to objects which are farther away. A sample image produced by the Kinect depth stream, as well as the image produced by the RGB camera for comparison, is shown in Figure 57.

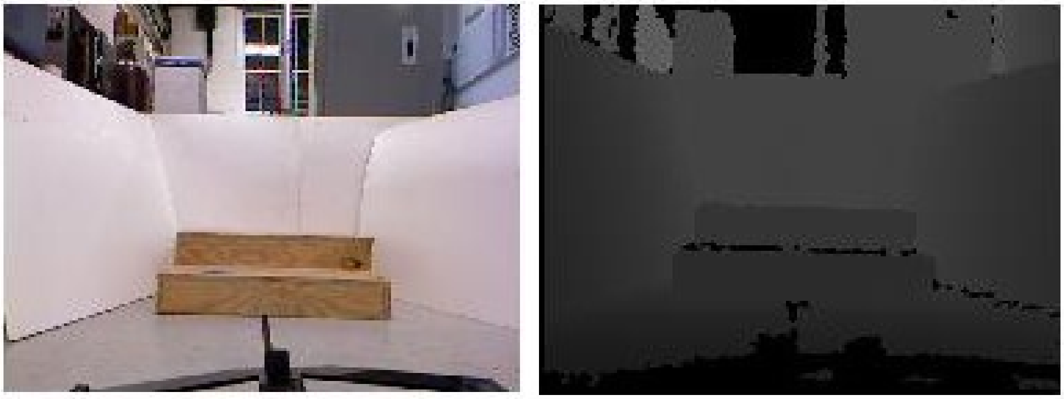


Figure 57 Kinect Depth Sensor Image

In order to determine the distance from a detected object to the Kinect, a calibration must be performed to develop an algorithm for transforming the values of each pixel in the depth image into a physical distance measured from the Kinect to the object. This calibration was performed by reading the depth value output at the center of the image while a flat surface was place in front of the Kinect at measured distances from the lens. Through this method, a linear relationship between the Kinect depth stream output and the distance of objects from the Kinect was determined, which was used to convert the depth information from the Kinect into physical distances measured in inches.

Another calibration performed was the used to determine the vertical size of objects detected by the Kinect. An object of known size was placed at measured distances from the Kinect and the number of vertical pixels of the object's image were measured. Using these measurements, as well as the previous depth calibration, a linear relationship between number of pixels per inch and distance from the Kinect was determine, which then can be used to determine the height of any object within the calibrated distance. The linear relationships for both distance and height were determined for distances from two to six feet from the Kinect, which is sufficient for the scope of stair detection.

To determine the height and width of the step, the image processing algorithm first takes 90 different depth images over three seconds from the Kinect depth image stream and averages them together to create one image. This is done to reduce noise in the image, since a small amount of intermittent noise is contained in each depth image. Once this composite image is obtained, the image is cropped around the center of the image, reducing the vertical size by one half and the horizontal size by one third. The MATLAB Sobel edge detection algorithm is then used to analyze the image, with a threshold value of 3. The resulting image is shown in Figure 58.

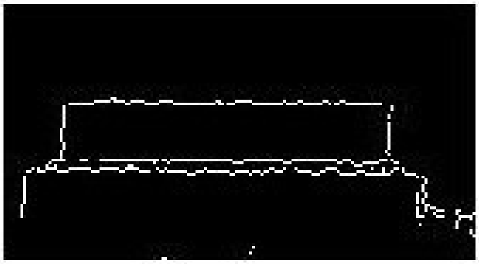


Figure 58 Stair Edge Detection Image

This image consists of binary pixel values, where each edge pixel has a value of one, while each non-edge pixel has a value of zero.

To find the top edge of the top step, the pixel values for each horizontal row are averaged. For horizontal lines this gives an average value closer to one, while nonhorizontal lines have an average value much closer to zero. Next each horizontal row with an average value greater than 0.3 is found, and the number of the row, or the height, is recorded. These values are then compared, and the lines corresponding to the top and bottom of the top step are determined by comparing the distance between adjacent horizontal lines, with the assumption being that the top and bottom edges of the step will be greater than 4 pixels apart.

Now that the height of the stair in pixels is know, as well as the location of the stair face in the cropped image, the height of the step is found using the previously described linear pixel to height relationship. The depth of the step beneath the top is then found and compared to the depth of the top step, which gives the step width.

4.5.2 Executing Behavior

The parameterized behavior was implemented completely on the Arduino microcontroller, which directly controls the actuator servos. This was accomplished using the Arduino programming language, which is based on C++, as well the Servo library provided as open source material by the Arduino Team [74]. This library provides all commands necessary for the Arduino to communicate with the servos.

Since the servos utilize PWM for input, each of the joint angles was converted to a pulse width, and each individual step was then determined as a set of pulse widths. Using this method, the behavior was transformed into a series of pulses which are transmitted to all servos simultaneously. Using this method, every servo receives a pulse signal every step, regardless if its angle is changing or remaining constant for this step. This ensures all servos maintain the torque required to maintain the desired joint angle.

To initialize the stair-climbing behavior, a serial signal is sent to the microcontroller containing three bits. The first bit executes the stair behavior, while the second and third bits communicate the height and width of the stair. Once the behavior is executed, no further input to the microcontroller is necessary from the laptop controller. With the described parameterization, the robot will climb stairs of any height and width without further sensor input, with the exception of the small distance sensors on the front of the robot.

To detect and climb stairs, the robot is first positioned in front of the stairs to be climbed, and the MATLAB control program is executed. The MATLAB then performs the previously discussed image analysis to determine the height and width of the stair and this information is then sent via the USB serial connection to the Arduino Microcontroller, along with a signal directing the microcontroller to execute the stair-climbing program. The microcontroller them implements the forward walking behavior until the front distance sensors detect the front of the stair. The robot then executes the parameterized stair-climbing behavior, using the height and width information provided by the MATLAB control program. The robot continues to climb stairs until no further stairs are detected, at which point the microcontroller instructs all servos to assume their zero position, effectively making the robot stand motionless.

4.6 Experimental Results

4.6.1 Experimental Setup

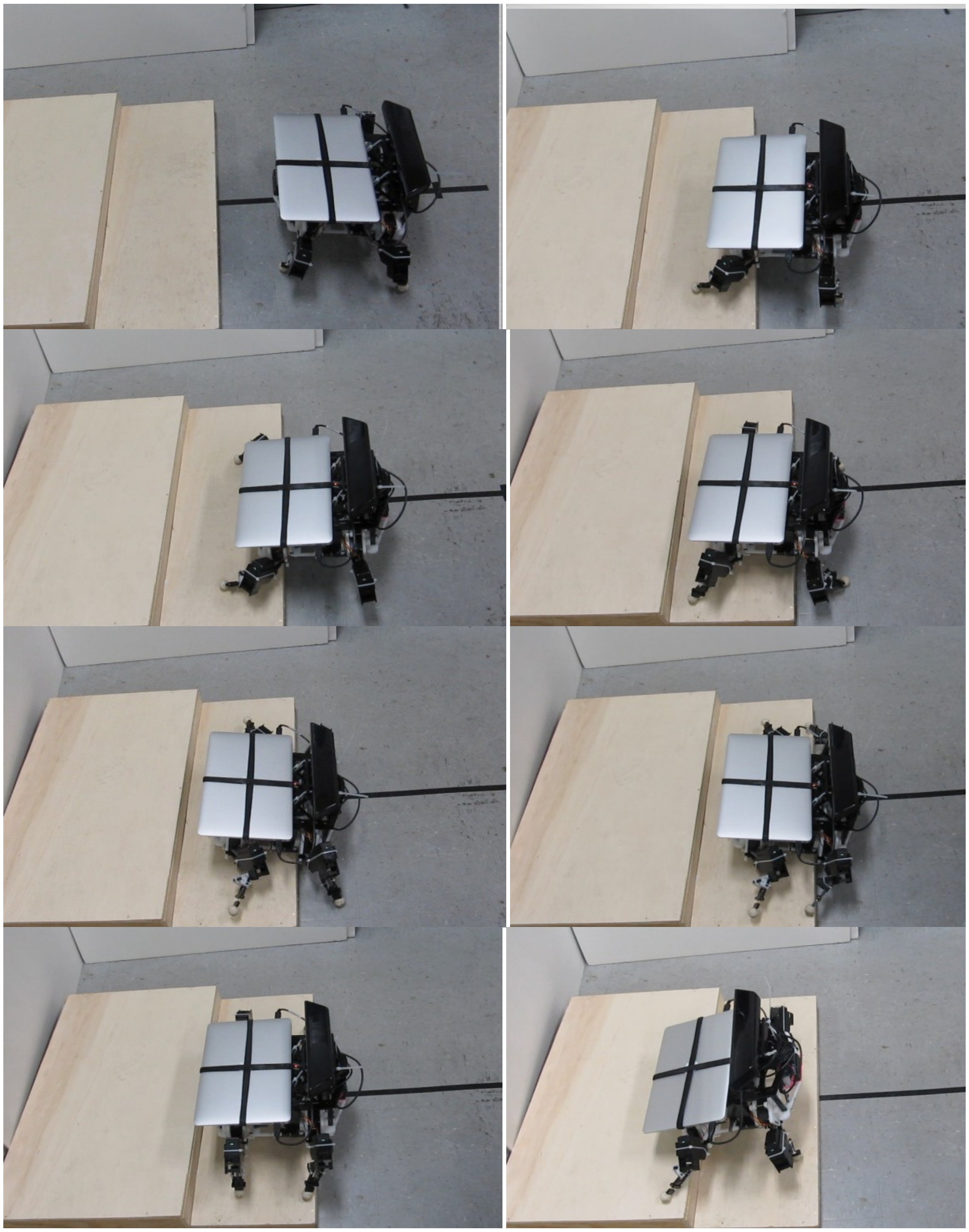
To test the ability of the Kinect sensor to detect stairs, the accuracy of the MATLAB program at analyzing the depth information to determine stair height and width information, the effectiveness of the parameterized behavior at climbing stairs of different heights, and the ability of the robot to climb stairs, the following series of experiments were performed.

Three different sizes of steps were constructed on which to test the robot, with heights of 2.5, 4, and 5 inches. The width of the 4 and 5 inch tall stairs is 10 inches, while the width of the 2.5 inch tall stair is 12 inches. Each step consists of two stairs, all with a top step width of 20 inches. The robot was then placed in front of each of these steps at a distance of 2.5 feet, and the MATLAB program was executed. The robot was then allowed to walk forward and climb the steps without any human intervention or input. In addition to this data, the height and width of the step determined by the MATLAB program was also recorded, to determine the accuracy of the height and width determination.

Also, to determine the sensitivity of the parameterized behavior to the accuracy of the stair input parameters, dummy values were substituted for the input parameters detected by the robot. These values were both for heights of one-half inch greater and less than the actual height of the stair, and for each it was the success or failure of the robot's climbing accept was recorded.

4.6.2 Results

Utilizing the parameterized behavior, along with the data from Kinect sensor, the robot successfully demonstrated the ability to climb all three stairs. These results are shown in Figures 59 through 62.



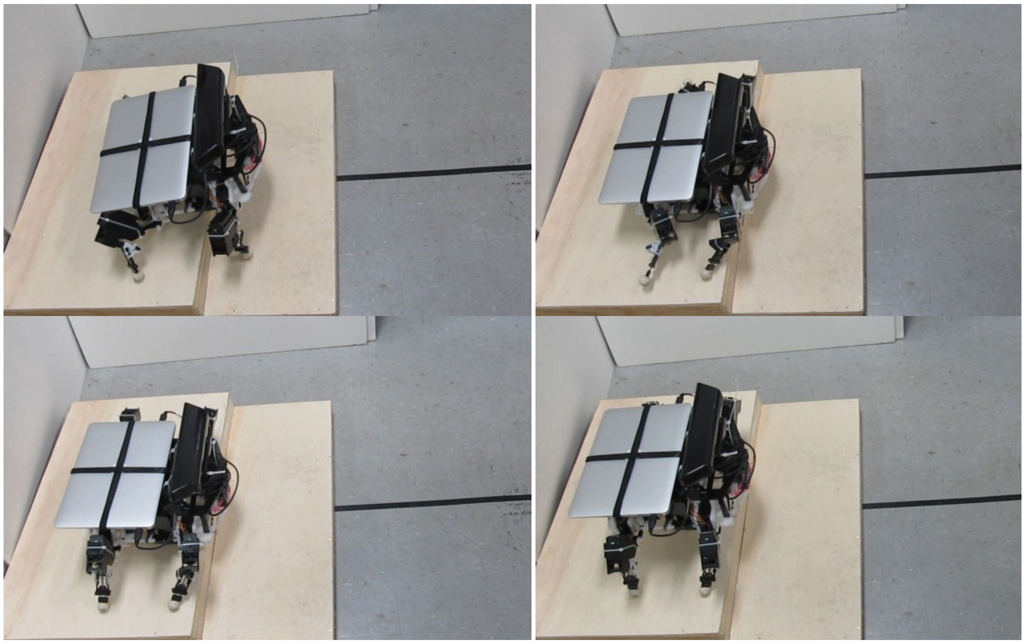
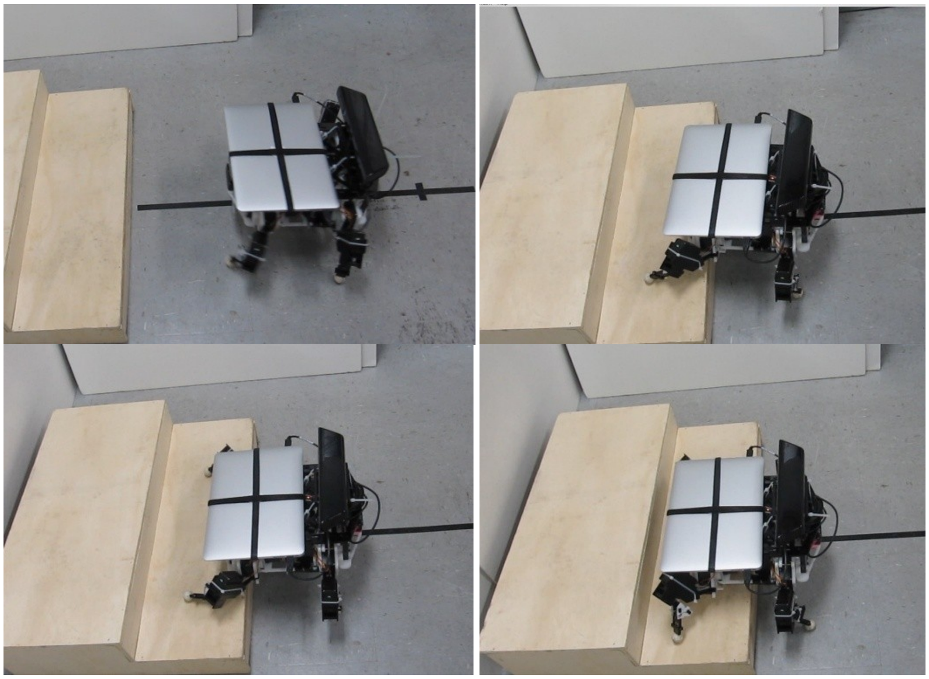


Figure 59 Robot Successfully Climbing 2.5 Inch Stair



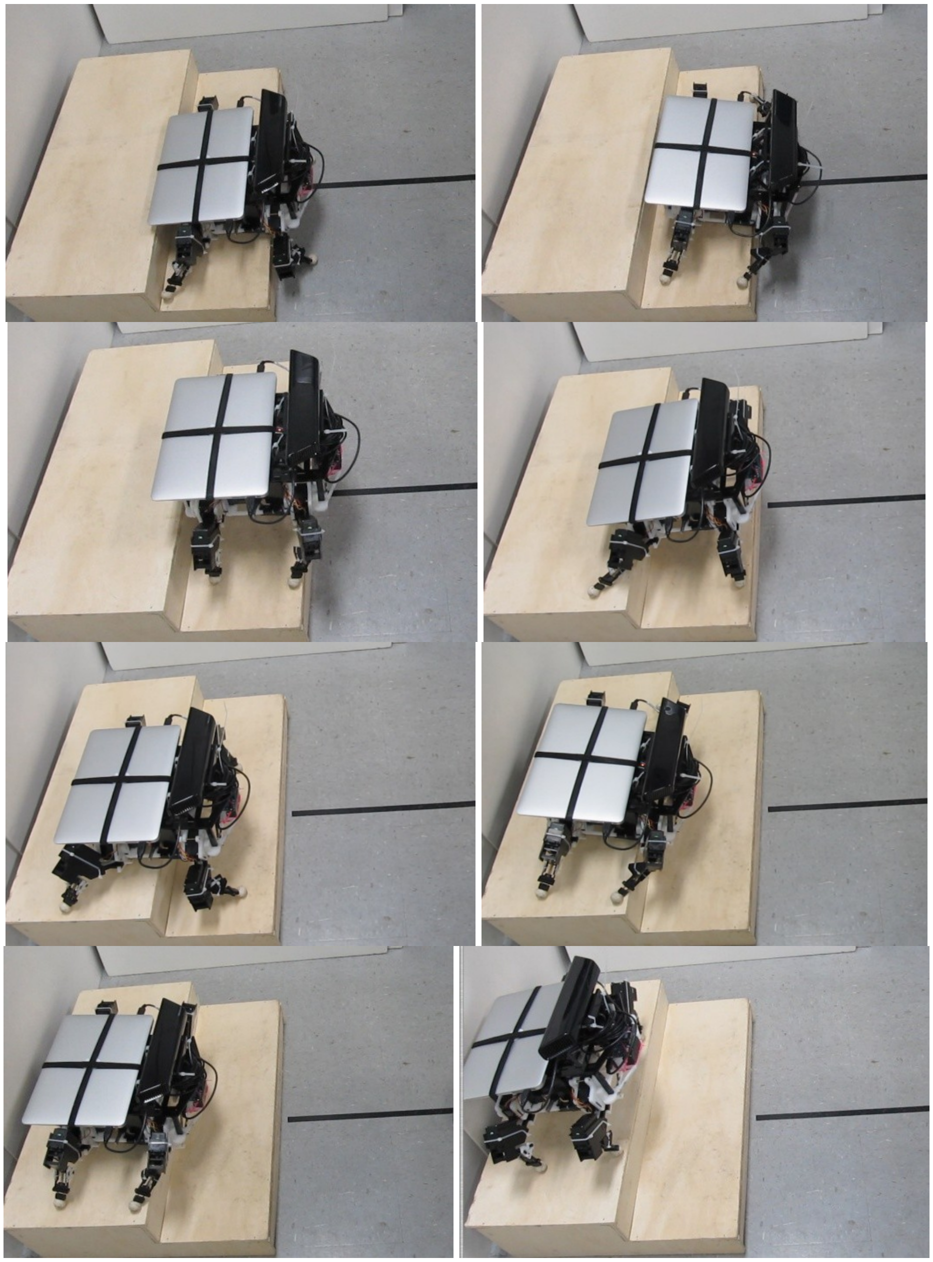
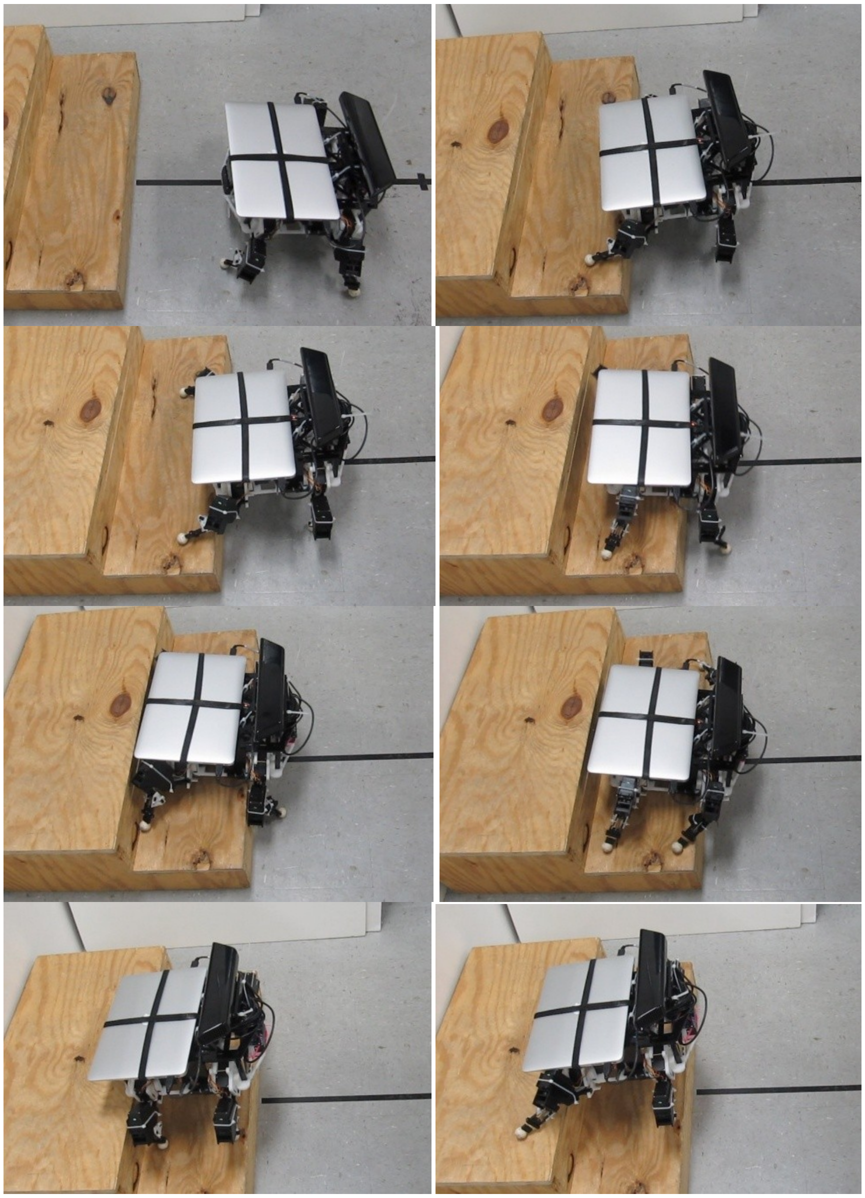


Figure 60 Robot Successfully Climbing 4 Inch Stair



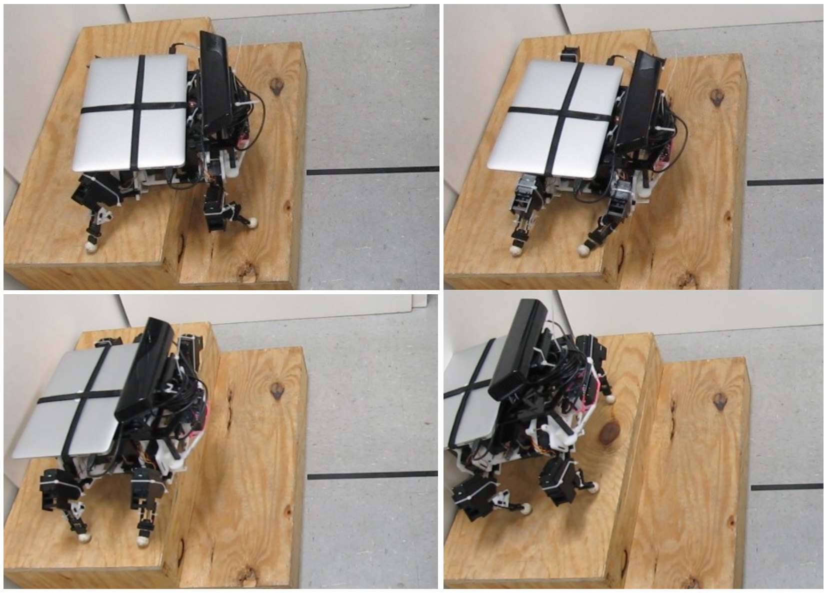


Figure 61 Robot Successfully Climbing 5 Inch Stair

In the figures above, it can be seen that the robot's configuration is the same for each stair, and that the robot successfully reaches the top step for each. No changes were made to the physical robot or the robot's programming between each stair, with only the MATLAB program and the Arduino being reset after each climb.

For the sensitivity test, the stair height parameter was changed to both 4.5 and 3.5 inches, and the robot then attempted to climb the 4 inch stair. For both of these attempts the robot successfully climbed the stair, despite the stair height parameter being both greater and less than the actual value.

4.7 Summary

The experimental results as shown in Section 4.6.2 demonstrate that both the parameterized behavior developed in the beginning of this chapter as well as the Kinect sensor implementation were both successful. The parameterized was shown to be capable of climbing stair of heights between 2.5 and 5 inches, without reconfiguration or re-programming. The Kinect sensor and MATLAB program demonstrated the ability to accurately detect and categorize stairs of multiple heights with a high degree of accuracy, without adjustment between experiments. And finally, the robot as a whole demonstrated the ability to detect and climb stairs of heights between 2.5 and 5 inches in height utilizing only onboard sensors, controllers, and power sources, the first demonstration of this accomplishment for a quadruped robot.

Chapter 5 – Conclusions

The goal of this project was to develop a quadruped robot with the largest possible payload capacity, which would be capable of climbing stairs of any height between 2.5 and 5 inches, while carrying the controllers and sensors necessary to detect the height and width of the stairs. As demonstrated in the previous chapter, the robot developed successfully met this goal, and the design was shown to be unique and capable.

5.1 Intellectual Contributions

The intellectual contributions of this work come from both the robot as a whole as well as the individual robot components.

1) A efficient and easily replicated design for a quadruped robot which has a demonstrated ability to both walk on flat terrain as well as climb stairs of several different sizes. While many previous stair-climbing quadrupeds exist, this robot design is unique in its ability to carry all sensors and controllers required to perform stair-climbing.
2) A parameterized behavior which is capable of adapting to stairs of various heights. While developed specifically for robots of the configuration of the robot developed in this work, the principles involved can be extended to quadrupeds of other configurations.
3) A highly accurate mobile robot sensor and control system utilizing commercially available physical components and software which is capable of detecting and classifying stairs of variable heights. This system could be

applied to any mobile robot capable of carrying the sensor and controller, including wheeled, tracked, or legged platforms.

5.2 Anticipated Benefits

There are several important concepts developed in this work will be beneficial to stair-climbing robots. First, the use of a parameterized behavior to climb stair of varying heights has broad applications to all legged mobile robots which are capable of stair-climbing. While many legged robots utilize complicated control systems to actively control their motion, a parameterized behavior could successfully be utilized to climb stairs for robots of many different configurations.

Another benefit of the parameterized stair-climbing behavior is the small amount of information needed to accurately characterize and climb stairs of varying heights. With a parameterized behavior, the only input required to modify the behavior for a particular stair configuration is the height and width of the step, which reduces both the sensor data required as well as the amount of computational resources needed to calculate the required behavior.

One final benefit is the demonstration of the utilization of the Microsoft Kinect as a robust and capable sensor that is well-suited to the task of stair-detection. With its relatively low weight and the ability to provide a three-dimensional model of the environment in close to real-time, the Kinect sensor could be utilized on a variety of stair-climbing platforms, including wheeled, tracked, and most legged configurations.

5.3 Future Work

While this robot has shown a capability for robust stair-climbing, its potential has not been fully reached, and there still remain many areas of future work which can both improve on the current design and methods as well as investigate new applications for the techniques developed in this effort.

5.3.1 Improved Actuator Performance

While the robot as developed in this thesis has demonstrated the ability to climb stairs with variable heights, with improved actuators capable of providing greater torque, the robot would be capable of operating with a long leg length, which would allow it to climb stairs of greater height or carry a greater payload. While the actuators currently used are the best analog servos currently available, it may be possible to incorporate digital servos with higher torque along with higher power batteries. This remains to be investigated.

5.3.2 Kinect Data Processing Improvement

While the Kinect was shown to be capable of detecting and classifying stairs, it should be possible to develop the stair-recognition more fully, to include creating a three-dimensional model of the stair, which would allow the robot to more intelligently detect and classify the stair. This process could also be used to have the robot seek out stairs to climb in an unknown environment, removing the need for the robot to be placed in a specific location in front of the stair.

5.3.3 Behavior Development and Parameterization for Small Stairs

Using the current parameterized behavior, it has been demonstrated that the robot is capable of climbing stairs with heights of greater than 2.5 inches. To accommodate smaller stairs, a different parameterized behavior should be developed, which would allow the robot to utilize a faster, more efficient behavior to traverse the smaller stairs. This type of behavior could also be generalized to accommodate rough terrain or slopes, since the geometry of these terrains is similar.

Bibliography

[1] J. Daintith, *A Dictionary of Computing*. Oxford University Press, Inc. New York, NY, USA, 2008.

[2] B. W. Spranklin, “Design, analysis, and fabrication of a snake-inspired robot with a rectilinear gait,” 2006.

[3] G. Figliolini and M. Ceccarelli, “Climbing stairs with EP-WAR2 biped robot,” in *IEEE International Conference on Robotics and Automation, 2001. Proceedings 2001 ICRA*, 2001, vol. 4, pp. 4116- 4121 vol.4.

[4] G. Figliolini, M. Ceccarelli, and M. Di Gioia, “Descending stairs with EP-WAR3 biped robot,” in *2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2003. AIM 2003. Proceedings*, 2003, vol. 2, pp. 747- 752 vol.2.

[5] P. Michel, J. Chestnut, S. Kagami, K. Nishiwaki, J. Kuffner, and T. Kanade, “GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007*, 2007, pp. 463-469.

[6] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, “The intelligent ASIMO: system overview and integration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002*, 2002, vol. 3, pp. 2478- 2483 vol.3.

[7] “Humanoid artificial people | Buskonia.com | Curiosities, Leisure, Search, Health, humor and varieties.” [Online]. Available: http://www.buskonia.com/en/futuro/humanoides-y-gente-artificial/. [Accessed: 23-Oct-2011].

[8] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, D. M. Helmick, and L. Matthies, “Autonomous stair climbing for tracked vehicles,” *The International Journal of Robotics Research*, vol. 26, no. 7, p. 737, 2007.

[9] S. D. Herbert, A. Drenner, and N. Papanikolopoulos, “Loper: A quadruped-hybrid stair climbing robot,” in *IEEE International Conference on Robotics and Automation, 2008. ICRA 2008*, 2008, pp. 799-804.

[10] M. Eich, F. Grimminger, and F. Kirchner, “A Versatile Stair-Climbing Robot for Search and Rescue Applications,” in *IEEE International Workshop on Safety, Security and Rescue Robotics, 2008. SSRR 2008*, 2008, pp. 35-40.

[11] M. M. Dalvand and M. M. Moghadam, “Stair Climber Smart Mobile Robot (MSRox),” *Autonomous Robots*, vol. 20, pp. 3-14, Feb. 2006.

[12] “US NSF - Robots For Real - The A-Team of Robots.” [Online]. Available: http://www.nsf.gov/news/special_reports/eng_mill/robots/ateam/. [Accessed: 24-Oct-2011].

[13] “ASGUARD | Robot Living.” [Online]. Available: http://www.robotliving.com/robot-news/asguard/. [Accessed: 24-Oct-2011].

[14] “Mohsen Moradi Dalvand - Publications Web Page.” [Online]. Available: http://www.mrdv.org/publications.html. [Accessed: 24-Oct-2011].

[15] C. Theeravithayangkura, T. Takubo, Y. Mae, and T. Arai, “Stair recognition with laser range scanning by limb mechanism robot “ASTERISK“,” in *IEEE International Conference on Robotics and Biomimetics, 2008. ROBIO 2008*, 2009, pp. 915-920.

[16] E. Z. Moore, D. Campbell, F. Grimminger, and M. Buehler, “Reliable stair climbing in the simple hexapod ‘RHex’,” in *IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA ’02*, 2002, vol. 3, pp. 2222-2227.

[17] Shuan-Yu Shen, Cheng-Hsin Li, Chih-Chung Cheng, Jau-Ching Lu, Shao-Fan Wang, and Pei-Chun Lin, “Design of a leg-wheel hybrid mobile platform,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*, 2009, pp. 4682-4687.

[18] Ke Jung Huang, S.-C. Chen, Ya Cheng Chou, S.-Y. Shen, Cheng-Hsin Li, and Pei-Chun Lin, “Experimental validation of a leg-wheel hybrid mobile robot Quattroped,” in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2976-2977.

[19] Chih-Chung Ko, Shen-Chiang Chen, Cheng-Hsin Li, and Pei-Chun Lin, “Trajectory planning and four-leg coordination for stair climbing in a quadruped robot,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 5335-5340.

[20] Shen-Chiang Chen, Ke Jung Huang, Cheng-Hsin Li, and Pei-Chun Lin, “Trajectory planning for stair climbing in the leg-wheel hybrid mobile robot quattroped,” in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1229-1234.

[21] S. Hirose, K. Yoneda, K. Arai, and T. Ibe, “Design of prismatic quadruped walking vehicle TITAN VI,” in *, Fifth International Conference on Advanced*

Robotics, 1991. "Robots in Unstructured Environments", 91 ICAR, 1991, pp. 723-728 vol.1.

[22] S. Hirose and K. Kato, "Study on quadruped walking robot in Tokyo Institute of Technology-past, present and future," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, 2000, vol. 1, pp. 414–419.

[23] S. Hirose, "A study of design and control of a quadruped walking vehicle," *The International Journal of Robotics Research*, vol. 3, no. 2, p. 113, 1984.

[24] S. Hirose, Y. Fukuda, and H. Kikuchi, "The gait control system of a quadruped walking vehicle," *Advanced robotics*, vol. 1, no. 4, pp. 289–323, 1986.

[25] "cyberneticzoo.com□ » Blog Archive□ » 19789 – PV-II 4-Legged Walking Machine – Hirose & Umetani (Japanese)." [Online]. Available: http://cyberneticzoo.com/?p=4001. [Accessed: 23-Oct-2011].

[26] "Hirose Fukushima Lab." [Online]. Available: http://www-robot.mes.titech.ac.jp/robot/walking/titan6/titan6_e.html. [Accessed: 19-Oct-2011].

[27] D. J. Pack and A. C. Kak, "A simplified forward gait control for a quadruped walking robot," in *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems '94. "Advanced Robotic Systems and the Real World", IROS '94*, 1994, vol. 2, pp. 1011-1018 vol.2.

[28] D. J. Pack, "Perception-based control for a quadruped walking robot," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, 1996, vol. 4, pp. 2994–3001.

[29] "Vision-Based Quadruped Walking Robot System." [Online]. Available: https://engineering.purdue.edu/RVL/Research/Roach/index.html. [Accessed: 20-Oct-2011].

[30] M. Buehler, R. Battaglia, A. Cocosco, G. Hawker, J. Sarkis, and K. Yamazaki, "SCOUT: a simple quadruped that walks, climbs, and runs," in *1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings*, 1998, vol. 2, pp. 1707-1712 vol.2.

[31] S. Talebi, M. Buehler, and E. Papadopoulos, "Towards dynamic step climbing for a quadruped robot with compliant legs," in *3rd Int. Conf. on Climbing and Walking Robots, Madrid, Spain*, 2000.

[32] I. Poulakakis, J. A. Smith, and M. Buehler, "Modeling and Experiments of Untethered Quadrupedal Running with a Bounding Gait: The Scout II Robot," *The International Journal of Robotics Research*, vol. 24, no. 4, pp. 239 -256, Apr. 2005.

[33] S. Talebinejad, “Compliant running and step climbing of the Scout II platform,” *M. Eng. Thesis, McGill University*, 2000.

[34] I. Poulakakis, J. A. Smith, and M. Buehler, *On the dynamics of bounding and extensions towards the half-bound and the gallop gaits*. Springer-Verlag, Tokyo, 2005.

[35] [Online]. Available: http://earth.gwu.ac.kr/html/Robots/node_s.html. [Accessed: 20-Oct-2011].

[36] “Ioannis Poulakakis / Research.” [Online]. Available: http://www.princeton.edu/~poulakas/Research/research.html. [Accessed: 21-Oct-2011].

[37] Bo Huang, Lining Sun, and Yufeng Luo, “Statically balanced stair climbing gait research for a hybrid quadruped robot,” in *Mechatronics and Automation, 2005 IEEE International Conference*, 2005, vol. 4, pp. 2067- 2071 Vol. 4.

[38] Bo Huang, Pengfei Wang, and Lining Sun, “Behavior-based Control of a Hybrid Quadruped Robot,” in *The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006*, 0-0 0, vol. 2, pp. 8997-9001.

[39] “Pei-Chun Lin’s Homepage - Research - Overview.” [Online]. Available: http://140.112.14.18/Homepage/research/r_overview.htm. [Accessed: 21-Oct-2011].

[40] J. Z. Kolter, M. P. Rodgers, and A. Y. Ng, “A control architecture for quadruped locomotion over rough terrain,” in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 811–818.

[41] J. R. Rebula, P. D. Neuhaus, B. V. Bonnlander, M. J. Johnson, and J. E. Pratt, “A Controller for the LittleDog Quadruped Walking on Rough Terrain,” in *2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 1467-1473.

[42] D. Pongas, M. Mistry, and S. Schaal, “A Robust Quadruped Walking Gait for Traversing Rough Terrain,” in *2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 1474-1479.

[43] A. Shkolnik, M. Levashov, I. R. Manchester, and R. Tedrake, “Bounding on rough terrain with the LittleDog robot,” *The International Journal of Robotics Research*, vol. 30, no. 2, pp. 192 -215, Feb. 2011.

[44] J. Buchli, M. Kalakrishnan, M. Mistry, P. Pastor, and S. Schaal, “Compliant quadruped locomotion over rough terrain,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*, 2009, pp. 814-820.

[45] K. Byl and R. Tedrake, “Dynamically diverse legged locomotion for rough terrain,” in *IEEE International Conference on Robotics and Automation, 2009. ICRA ’09*, 2009, pp. 1607-1608.

[46] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal, “Fast, robust quadruped locomotion over challenging terrain,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 2665–2670.

[47] A. Shkolnik and R. Tedrake, “Inverse Kinematics for a Point-Foot Quadruped Robot with Dynamic Redundancy Resolution,” in *2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 4331-4336.

[48] K. Byl, “Metastable legged-robot locomotion,” Massachusetts Institute of Technology, 2008.

[49] K. Byl, A. Shkolnik, S. Prentice, N. Roy, and R. Tedrake, “Reliable dynamic motions for a stiff quadruped,” in *Experimental Robotics*, 2009, pp. 319–328.

[50] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal, “Learning, planning, and control for quadruped locomotion over challenging terrain,” *The International Journal of Robotics Research*, vol. 30, no. 2, p. 236, 2011.

[51] “Boston Dynamics: Dedicated to the Science and Art of How Things Move.” [Online]. Available: http://www.bostondynamics.com/robot_littledog.html. [Accessed: 22-Oct-2011].

[52] “Computational Learning and Motor Control Lab | Research / Learning Locomotion.” [Online]. Available: http://www-clmc.usc.edu/Research/LearningLocomotion. [Accessed: 22-Oct-2011].

[53] J.-S. Gutmann, M. Fukuchi, and M. Fujita, “Stair climbing for humanoid robots using stereo vision,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings*, 2004, vol. 2, pp. 1407- 1413 vol.2.

[54] “Sony Global - Press Release - World’s First Running Humanoid Robot.” [Online]. Available: http://www.sony.net/SonyInfo/News/Press_Archive/200312/03-060E/. [Accessed: 23-Oct-2011].

[55] A. Albert, M. Suppa, and W. Gerth, “Detection of stair dimensions for the path planning of a bipedalrobot,” in *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2001. Proceedings*, 2001, vol. 2, pp. 1291-1296 vol.2.

[56] Yalin Xiong and L. Matthies, “Vision-guided autonomous stair climbing,” in *IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA ’00*, 2000, vol. 2, pp. 1842-1847 vol.2.

[57] J. Stowers, M. Hayes, and A. Bainbridge-Smith, “Altitude control of a quadrotor helicopter using depth map from Microsoft Kinect sensor,” in *2011 IEEE International Conference on Mechatronics (ICM)*, 2011, pp. 358-362.

[58] P. Benavidez and M. Jamshidi, “Mobile robot navigation and target tracking system,” in *2011 6th International Conference on System of Systems Engineering (SoSE)*, 2011, pp. 299-304.

[59] K. Hauser, T. Bretl, J. C. Latombe, K. Harada, and B. Wilcox, “Motion planning for legged robots on varied terrain,” *The International Journal of Robotics Research*, vol. 27, no. 11-12, p. 1325, 2008.

[60] “IBC Stairs Code.” [Online]. Available: http://www.amezz.com/ibc-stairs-code.htm. [Accessed: 30-Oct-2011].

[61] “Servo Motor Control - The Servo Motor | PyroElectro - News, Projects & Tutorials.” [Online]. Available: http://www.pyroelectro.com/tutorials/servo_motor/servomotor.html. [Accessed: 31-Oct-2011].

[62] “HS-805BB - Hitec RCD.” [Online]. Available: http://www.hitecrcd.com/products/analog/giant-scale/hs-805bb.html. [Accessed: 31-Oct-2011].

[63] “How to Hack the Microsoft Kinect - Overview.” [Online]. Available: http://liambean.hubpages.com/hub/How-to-Hack-the-Microsoft-Kinect-Overview. [Accessed: 01-Nov-2011].

[64] “Tenergy NiMh 12V 2000mAh Battery Pack w/Bare Leads.” [Online]. Available: http://www.tenergy.com/12V-2000-mAh-NiMH-Battery-Pack-with-Bare-Leads?sc=59&category=38221. [Accessed: 02-Nov-2011].

[65] “kinect/Tutorials/Adding a Kinect to an iRobot Create - ROS Wiki.” [Online]. Available: http://www.ros.org/wiki/kinect/Tutorials/Adding%20a%20Kinect%20to%20an%20iRobot%20Create. [Accessed: 02-Nov-2011].

[66] “Toko-Robot.Com.” [Online]. Available: http://toko-robot.com/shop/product_info.php?products_id=207. [Accessed: 02-Nov-2011].

[67] "Tenergy Premium NiMH C 5000mAh Rechargeable Battery." [Online]. Available: http://www.tenergy.com/10208. [Accessed: 01-Nov-2011].

[68] "Arduino - ArduinoBoardMega2560." [Online]. Available: http://arduino.cc/en/Main/ArduinoBoardMega2560. [Accessed: 31-Oct-2011].

[69] "Apple - MacBook Air - Compare the 11- and 13-inch MacBook Air." [Online]. Available: http://www.apple.com/macbookair/specs.html. [Accessed: 02-Nov-2011].

[70] "McMaster-Carr - Delrin (Made with Acetal Resin)." [Online]. Available: http://www.mcmaster.com/#delrin-(made-with-acetal-resin)/=eqv87s. [Accessed: 01-Nov-2011].

[71] "Acetal Sheet - Delrin® (Thermoplastic Polymer): TAP Plastics." [Online]. Available: http://www.tapplastics.com/shop/product.php?pid=337. [Accessed: 01-Nov-2011].

[72] "VLS 3.60 - Universal Laser Systems." [Online]. Available: http://www.ulsinc.com/products/vls360/. [Accessed: 01-Nov-2011].

[73] "Kinect Microsoft SDK - File Exchange - MATLAB Central." [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/33035-kinect-microsoft-sdk. [Accessed: 13-Nov-2011].

[74] "Arduino - HomePage." [Online]. Available: http://arduino.cc/en/. [Accessed: 13-Nov-2011].