uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Ruba Kayyali

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.A.Sc. (Electrical and Computer Engineering)

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

A Virtual Haptic Motor Rehabilitation System

TITRE DE LA THÈSE / TITLE OF THESIS

Shervin Shirmohammadi

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

Abdulmotaleb El-Saddik

Peter X. Liu

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

# A Virtual Haptic Motor Rehabilitation System

by

Ruba Kayyali

A thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements for the degree of

Master of Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

*Your file Votre référence*
*ISBN:* 978-0-494-73860-3
*Our file Notre référence*
*ISBN:* 978-0-494-73860-3

# Abstract

There are over 40,000 strokes in Canada each year, with a 75% rate of physical impairment. Virtual Rehabilitation was proposed as an alternative for recovering stroke patients, since conventional rehabilitative methods limit patients with time constraints. Our research presents a virtual rehabilitation system, consisting of a series of applications and using the Immersion CyberForce® system. The system is designed to improve patients' abilities to perform everyday tasks, such as drinking tea, as per the feedback of occupational therapists. It also includes an interface for occupational therapists to configure each exercise. We designed a progress measurement component for automated advancements without requiring an occupational therapist to analyze performance results. An LSR Analyzer normalizes the raw sensor values read from the Immersion CyberGlove, for better sensor readings to be fed into an Artificial Neural Network. This is used to minimize the requirements of the calibration process. Analysis of all results and feedback is presented.

# Acknowledgements

I would like to thank my Professor and Advisor, Dr. Shervin Shirmohammadi for the opportunity to work on such an exciting and current research project. I would like to thank him for all his time, energy and support.

I would also like to portray my gratitude to Dr. Abdulmotaleb El Saddik and Dr. Edward Lemaire for their assistance throughout my research work.

I owe an undeniable debt of gratitude to my parents and family, for raising me to be the person I am today, and for all their help and support, and for their perseverance in providing me with the highest standards of education.

And last, but not least, I cannot forget all the late hours and volunteering by Hani Jabbour. I thank you deeply for your support and for your everlasting presence in my life.

# Table of Contents

# List of Figures

# List of Tables

# List of Definitions

- **VR – Virtual Rehabilitation**: Virtual rehabilitation is the patient's training based entirely on, or augmented by virtual reality simulation exercises. [4]
- **LSR – Least Squares Regression**: Regression analysis method which minimizes the sum of the square of the error as the criterion to fit the data. This can refer to linear or curvilinear regression. [52]
- **MCP – Metacarpal**: Any bone of the hand between the wrist and the fingers. [56]
- **DoF – Degrees of Freedom**: Degrees of freedom (DOF) are the set of independent displacements and/or rotations that specify completely the displaced or deformed position and orientation of the body or system. [53]
- **ANN – Artificial Neural Network**: a mathematical model or computational model that tries to simulate the structure and/or functional aspects of biological neural networks. [51]
- **Palmar**: Relating to the palm of the hand or the sole of the foot. [57]
- **Dorsal**: Belonging to or on or near the back or upper surface of an animal or organ or part. [58]
- **Occupational Therapist:** (OT) is a therapist who is trained in the practice of occupational therapy. [59]

# Chapter 1

# Introduction

There are between 40,000 to 50,000 strokes in Canada each year, an average of one stroke every ten minutes.[60] A stroke is the rapid loss of brain function due to a change in the blood supply to the brain. As the result of a stroke, the patient can lose the ability to move body limbs, to understand or formulate speech, or even loss in vision. [61] Up to 75% of stroke patients will continue to live with some varying degree of severity of impairment. The neurological damage caused by a stroke is usually permanent, but can be reduced with treatments including medication and supportive care. Strokes can also affect patients on the emotional and mental levels. The patients' resulting physical impairments and the following frustrations in adapting to their new limitations can increase the emotional and mental toll of the patients' stroke experience. The stroke itself could have damaged the emotional centers within the brain.

In order for a patient to regain function in any impaired limb, rehabilitative exercises are needed. This is where stroke rehabilitation is needed. Stroke rehabilitation is the process of treating stroke patients to regain and re-learn the skills lost due to stroke-related impairments [61]. Its goal is to help the patients live as normal a life as possible, and to learn to adapt to their new limitations. Occupational therapy and physical therapy play lead roles in stroke rehabilitation. Their roles overlap in certain areas, but emphasis for physical therapy is in re-learning the gross motor functions, while occupational therapy aims at everyday activities, more commonly referred to as the "Activities of daily living" (ADL's). These activities include eating, drinking, bathing, dressing, etc.

Occupational therapists follow a predefined process to perform the necessary sequence of actions in determining the course of treatment for stroke patients. Once the goals have been set, and the patient's limitations defined, the occupational therapists work closely with the patients on implementing the course of treatment. The occupational therapist has to be flexible enough to adapt to the requirements of each

patient, and be capable of providing prolonged periods of time to the treatment of each patient. The conventional rehabilitative systems that occupational therapists employ require the constant monitoring and assistance of the occupational therapist. These conventional methods provide a recovering stroke patient with an average of one to two half hour sessions a day with an occupational therapist [1]. The time limitation hinders the recovery of the stroke patients as it is far from sufficient for a speedy recovery. In order to increase the efficiency of the rehabilitative process, more frequent and repetitive practice must be made accessible to the patients. This means that either more occupational therapists have to be introduced into the field of physical rehabilitation, or fewer patients have to be treated. Either option is not a viable solution. Amongst other proposals, Virtual Rehabilitation was introduced over a decade ago, as an alternative for recovering stroke patients.

Virtual Rehabilitation can be used to design occupational therapy systems, using specific hardware machines, and tailored virtual exercises. The introduction of Virtual Reality into the process of rehabilitation is not a novel idea. It was a natural evolution of incorporating virtual reality to replace the more conventional methods. The use of Virtual Rehabilitation does not require the constant monitoring and attention of an occupational therapist. This means that time will no longer be a limiting factor. The use of Virtual Rehabilitation also meant that more devices could be designed in a form that would eventually allow patients to practice and go through their rehabilitative exercises from the comfort of their own homes.

## 1.1. Motivation

To enable patients to perform more exercises and to address the requirements of rehabilitation, Virtual Reality (VR) technology with animated exercises can be used to allow the patients to practice on practically any computer, even at the convenience of their own home [2-3]. Virtual Rehabilitation can provide recovering stroke patients with the ability to perform rehabilitation exercises with more frequency and repetitiveness as was previously provided by occupational therapists.

Virtual Rehabilitation is a means of restoring the health of a patient while relying, entirely or partially, on the aid of virtually simulated environments. The rehabilitative process of a patient could be a form of conventional therapy augmented with virtual rehabilitation, or it could be comprised of only virtual reality-based therapy. In 2002, Professor Daniel Thalmann of EPFL in Switzerland and Professor Grigore Burdea of Rutgers University in the USA coined the term "Virtual Rehabilitation", applying it to both physical therapy and a form of counseling in cognitive psychology called cognitive interventions. [4] Since then, the world of Virtual Rehabilitation continued to grow at a considerable rate. In 2003, a group of researchers from the University of Ulster in Ireland lent their rehabilitative science and computer engineering skills to the Royal Victoria Hospital in Belfast to establish one of the first studies into the effects of Virtual Rehabilitation on the speed of recovery of stroke patients. [5] After inventing the Rutgers Master II-ND haptic glove, researchers from the Rutgers University in the USA went on to create a more accessible version of a Virtual Rehabilitation system through the use of the Microsoft Xbox gaming consoles in 2006. Using the low-cost Microsoft Xbox and Essential Reality's P5 Gaming Glove, the researchers were able to create two types of finger-flexing exercises by applying minor modifications to the equipment and writing the necessary software. [6]

At the same time, the haptic technology or haptics has been recently employed in many Virtual Rehabilitation applications. Haptic, which is derived from the Greek verb "haptesthai" meaning "to touch", refers to the science of touch and force feedback in human-computer interaction. The haptic technology provides an interface to the human user through the sense of touch, by applying various motions and senses to the human user. This simulation of realistic motions has been applied in the field of virtual reality to provide the user with the ability to manipulate objects in the virtual environment and to sense the reactions of motions on these objects.

Haptic technology has been available since the 1950's, when the first teleoperators were built by Raymond Goertz, at the Argonne National Laboratory in the United States. Those teleoperators were electrically actuated and used to handle

radioactive substances. [7]. However, the earliest use of haptic feedback was in the control systems of large aircrafts that were operated using servo systems. The haptic device used in such big aircrafts provided the "shaking" feature of the control system that signaled to the pilot that the aircraft's angle has approached the critical stall point. This was put in place to compensate for the lack of shaking in the more advanced control systems that were operated by the servo systems. Since this first haptic device, the application of haptic feedback has encompassed many fields including gaming, medicine, robotics and actuators.

One of the most significant applications of haptic technology is in the field of medicine. By providing realistic and hands-on experience to medical students, the use of haptic technology has revolutionized medical education. From applications in surgery and diagnosis, to rehabilitation and training, haptic devices are beginning to take center stage in the medical field. Looking at the portfolio of products of one of the leading manufacturers of haptic devices, Immersion Corporation, a variety of systems can be found that aid medical professionals in several surgical procedures. These include laparoscopic and endoscopic systems, as well as an arthroscopic surgical simulator. Immersion also provides several systems that can aid medical students in learning how to administer intravenous catheters to patients, in a virtually simulated environment using haptic feedback [8]. Another major field of haptic feedback application is in the gaming industry. Since the 1980's, arcade games such as car racing simulators, have included the use of haptic feedback. These applications were intended to provide a more realistic and enjoyable experience to the gamer, through the addition of realistic collision motions. The Immersion Corporation's implementation of haptic force feedback in the arcade games include transmissions of the road conditions through the steering wheel, as well as vibrations and torque feedback and realistic simulations of off-road driving [9].

Haptic-based virtual rehabilitation offers the potential to create systematic human testing, training and treatment environments that allow precise control of complex dynamic 3D stimulus presentations, behavioral tracking, performance measurement,

data recording, and analysis [10]. Yet, haptic-based virtual rehabilitation has only been made available within the past few years. This is mainly due to the difficulties in the design of compact haptic devices that are reliable enough and provide a high grade of accuracy. [11] However, with the advances in research in the area of haptic devices, they are evolving and getting cheaper, more flexible, and more compact in size. Certain devices, such as the Immersion CyberForce system, shown in Figure 1, have the ability to measure the user's hand movements within a defined virtual environment, and to track his/her arm movements within a limited workspace area. Such a device can provide a rehabilitative system with the ability to measure a patient's precision in controlling his/her arm's motions and grasp of his/her hand. This implies that the incorporation of haptics in virtual reality based rehabilitation systems opens up many applications and possibilities, such as haptic guidance and augmented feedback. It further allows the creation of automated progress measurement systems that can analyze and detect the effectiveness of the rehabilitative process on a patient's recovery. This automated means of monitoring a patient's progress would also reduce the time needed by an occupation therapist, allowing for an even faster recovery time.

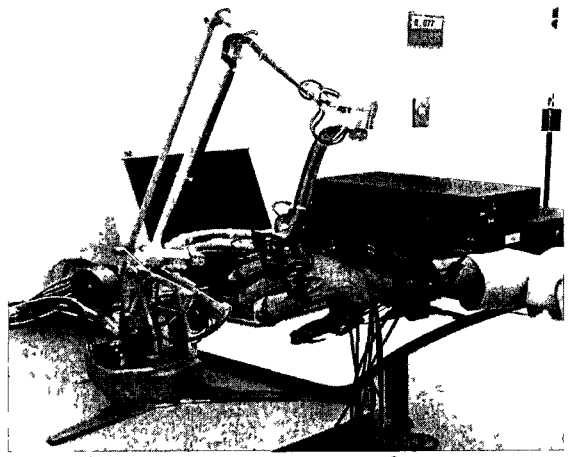

Figure 1: Immersion's CyberForce® System

## 1.2. Research Problems

There have been several implementations for haptic-based virtual rehabilitation systems. Such systems have managed to incorporate intensive exercises that are designed to target the patient's motor and cognitive skills, in a repetitive and progressive manner. However, the majority of those exercises lack a familiarity with

the patients, even though they have been proven effective in rehabilitating stroke patients. Furthermore, there continues to be a gap between the design of these systems and the actual integration of such a system with real rehabilitation patients. The main problem with these designs is the absence of the Occupational Therapist (OT). OTs need to be able to carefully configure the system in order to, for example, select the amount of time delay, angle, zoom, difficulty, and other parameters that will be used as a form of tracking of the patient's progress. This gives the OTs better control of the system, and allows them to determine the progress and improvements of a patient. Measurement of this progress and appropriate consequential actions is the key to the successful recovery of the patients. Moreover, using scientific methods to measure this progress means that a patient can advance at a quicker and more precise rate. Earlier methods relied on the OT's expertise to determine if a patient is capable of advancing to more complicated exercises. This method could introduce a delay in a patient's progress, as well as errors in judgment on behalf of the OT's.

Other problems included the actual hardware itself. The use of such complex haptic devices meant that patients had to sit through complicated and tiresome calibration processes that were sometimes difficult to do, depending on the patient's physical abilities. The calibration process requires the patient to have complete control over his/her fingers and hand, which is usually the problem the patient, is suffering from. The problem with the calibration of the haptic devices also means that the sensor readings are not rendered correctly in the virtual environment, where the movements of the hand in the virtual environment do not always replicate the actual physical movements.

## 1.3. Approach

The proposed system in this thesis is a Haptic Virtual Rehabilitation system whose goal is to assist patients who suffered from a stroke, acquired brain injury, muscular sclerosis, paraplegia and upper extremity disabilities. The system allows patients to perform daily-life exercise that they need to do in real life, such as moving common objects, eating soup with a spoon, and other exercises. The system also

provides Occupational Therapists with the ability to configure the weight of the virtual objects in each exercise, as well as other features, and to monitor the progress of each patient, by correlating the results gathered from each exercise with base data gathered against predefined criteria. The system uses the CyberForce system from Immersion Corporation, shown in Figure 1. This specific hardware was chosen because it measures the position and orientation of each finger, the wrist, and the hand in general, and simulates objects' weight by exerting a downward force. The system's requirements were the result of several consultations with the Rehabilitation Center of the Ottawa General Hospital and the Discover Lab at the University of Ottawa. The initial implementation was also placed under the analysis of a group of five occupational therapists that provided thorough feedback and further amendments to the design of the system. As a result of these consultations and trials, the graphical user interface was modified to include further details, tailored for each specific exercise, as well as a new list of exercises derived from common daily activities. The system was then placed under test by 5 volunteers and their data collected and analyzed. The final aim of the project is to have the system placed under trial in the Ottawa General Hospital where patients can provide us with feedback and further recommendations for improving the system. The research then went on to tackle the new issues as was revealed by the occupational therapists. The process of configuring the CyberForce system was deemed too difficult for certain stroke patients, especially for those unable to open their hands and stretch their fingers. Another problem with the system was its rendering of the user's hand in the virtual environment, which did not seem to replicate the actions of the user in a precise manner. This brought about the need create a component that would enhance the data read from the sensors and produce a more precise rendering within the virtual environment. The reason behind this lack of precision is that the CyberGlove has a total of 22 sensors placed at critical points, intended to measure the movements of the joints of the user's hand. Due to variations of the human hand size, the relative positions of the sensors to their corresponding critical points vary from one user to another. This meant that the default calibration mechanism that came with the glove was insufficient to accurately capture the calibration values needed to effectively use the glove. To resolve this, a neural network was designed that would be trained to

produce calibration data given the dimensions of the user's hand. The dimensions themselves were acquired rapidly using a two dimensional image of the user's hand, set against a chessboard, and processed to get the user's hand segment sizes, which are then inputted into the Neural Network. The design of the neural network was detailed in [12]. The use of the Artificial Neural Network (ANN) to calibrate the CyberGlove system was a result of its previous success in the calibrations of many complex systems and its effectiveness in learning the underlying input/output patterns [13] [14] [15] [16]. The ANN was retrained with different input/output combinations that were missing in [17], such as hand width, and also extended the method to measure the thumb and abduction related sensors, which is physically different and more difficult from the other fingers.

The outputs are collected using the manual calibration process. However, since the training of the ANN relies on the use of only a few input sets of data, it becomes more difficult to find the best suited gain and offset values for the sensors. For this purpose, the system was enhanced to use the Least Square Regression method for estimating the parameters [18]. The Least Square Regression method is used to approximate the solution of over determined systems, where the number of equations exceeds the unknown variables [19]. The Least Square Regression method is applied in two steps. The first requirement is to be able to produce the gain and offset values for each sensor of the CyberGlove system. These sensors include the flexion and the abduction sensors. The second application of the Least Square Regression method is to determine the cross-coupling effect of adjacent flexion sensors to the abduction angles in between those sensors.

## 1.4. Thesis Contributions

The thesis presented here has made several research contributions to the field of haptic-based virtual rehabilitation, including:

1. Design and development of realistic exercises, based on every-day activities that a patient would strive to be able to perform on his/her own ability. These exercises

were designed to help the patient build his/her own independence and aid in a faster recovery from a stroke inflicted disability in his/her upper extremities.

2. Design and development of an automated progress measurement system that would monitor the advancements that a patient makes as (s)he uses the system. The data is analyzed automatically and the user will be presented with the results. The user will also be instantly informed of the progress and the patient can advance to a higher difficulty level autonomously.
3. Design and development of a simple-to-use method to more accurately and rapidly calibrate the CyberGlove instrument for a patient. The method is especially practical for patients with severe disabilities, as it requires a simple image of the user's hand to be taken and requires no physical activity or the current standard calibration process of the device, which can be strenuous and in some cases impossible to do for such patients. It should be noted that this author's contribution to this part was the design and development of a component that would select the appropriate neural network for each sensor of the instrument, using Least Square Regression (LSR), as will be explained in chapter YY, to compare between the training data sets and the evaluation data collected from the manual calibration process.
4. Proof of concept and validation of the proposed design with human subjects.

## 1.5. Scholastic Achievements:

The following papers have been published as a result of this thesis work:

- R. Kayyali and S. Shirmohammadi, "A Benchmarked Automated Progress Measurement System for Haptic Motor Rehabilitation", International Journal of Advanced Media and Communication, Vol. 3, No.1, 2009, pp. 179 – 196.
- R. Kayyali, S. Shirmohammadi, A. El Saddik, "Measurement of Progress for Haptic Motor Rehabilitation", Proc. IEEE Workshop on Medical Measurement and Applications, Ottawa, Canada, May 9-10 2008, pp. 108 – 113.
- R. Kayyali, S. Shirmohammadi, A. El Saddik, and E. Lemaire , "Daily-Life Exercises for Haptic Motor Rehabilitation", Proc. IEEE Workshop on Haptic

Audio Visual Environments and Games, Ottawa, Canada, October 2007, pp. 118 – 123.

- R. Kayyali, A. Alamri, M. Eid, R. Iglesias, S. Shirmohammadi, A. El Saddik, and E. Lemaire, "Occupational Therapists' Evaluation of Haptic Motor Rehabilitation", Proc. IEEE Conference of Engineering in Medicine and Biology Society, Lyon, France, August 23-26 2007, pp. 4763-4766.

## 1.6. Thesis Outline

Chapter 2 will provide further information about related research and other state of the art research being carried out with regards to the field of Haptic-based Virtual Rehabilitation. A summary of related works will also be provided.

Chapter 3 will describe the realistic exercises that were designed based on everyday activities of patients. It will also detail the new configuration interface that provides occupation therapists with a more fluid use of the system. This chapter will also include the trial phase that was initially run with a number of volunteer occupational therapists and details their feedback for the system.

In chapter 4, we will explain the new automated progress measurement component that was added to the system. This component allows occupational therapists to follow the progress of patients at their own time, as well as provide a real-time progress report with automated level advancement for the patients. Again, a trial phase was carried out for the new component and an analysis of the data is included.

Chapter 5 will detail the practical calibration method, including the Artificial Neural Network design that was used based on the work in [12], and the design of the Least Square Regression component which enhances the raw sensor data values into a mapping that reflects the real angles of the human hand. Trial results and analysis of the implementation are also shown.

Finally, chapter 6 provides a conclusion of the work presented, as well as details about further work needed, and further research directions to be taken.

# Chapter 2

# Related Work

The use of VR systems has not been restricted to just physical rehabilitation, within the field of medicine, but has encompassed various fields including laparoscopic surgery, and even the psychological aspects of patients. In [20], the authors describe the use of VR-based leisure systems to elevate the self-esteem and independence of adults with cerebral palsy. Virtual reality systems have also been used in diagnostic medicine, where a VR system was developed and applied for simulation of cancer-related surgeries, diagnosing different forms of cancer, and even for a palliative care support system [21]. However, in the field of rehabilitative medicine, the use of VR systems has continued to grow.

The introduction of haptics into VR-based systems has lead way to the development of many new applications within the field of medical research. For example, [22] describes the application of haptic technology to create a surgery simulator for training surgeons in the field of minimally invasive surgery. Other applications of haptics in virtual rehabilitative systems include the Rutgers Ankle and the Rutgers Arm, both pioneered at the Rutgers University in New Jersey, U.S.A. The Rutgers Ankle is a robotic device that allows for orthopedic rehabilitation. The device allows for all three degrees of freedom of the patient's ankle, allowing for a full range of motion. The device was also coupled with a haptic interface, which provides a force feedback response to the patient's movements [23]. The system was also used in a tele-rehabilitation application, where a network of computers would allow the patients to undergo rehabilitative therapy from the comfort of their homes, but nevertheless with the help and monitoring of occupational therapists [24]. The results of this work, reported in [25] and [26], explains how some of the patients' ankle muscles capabilities were improved when introducing haptic effects using the Rutgers Ankle Rehabilitation Interface. As for the Rutgers Arm, the rehabilitative system comprised of a special table, a three dimensional tracker which measures the location of the arm within a

defined three dimensional space, a computer work station and a library of three dimensional virtual rehabilitative exercises. The Rutgers Arm was also analyzed and tried with an actual stroke patient and the improvements in the condition of the patient were noted in [27]. The rehabilitative system was then improved to produce the Rutgers Arm II. The system's tracking component was replaced from the Polhemus Fastrack magnetic tracker, by a custom-designed vision tracking system [28]. As for other upper extremity, several attempts have been made to incorporate haptics for arm and wrist motor function rehabilitation [29, 30, 31]. The first of these publications concentrates on the creation of robotic devices that can be deployed in motor rehabilitation systems, and the last paper proposes the use of ER actuators to simulate force feedback. However, there is still no emphasis on the ease of use of the system and its simplicity, from the perspective of the patient. In fields other than rehabilitative medicine, haptic-based virtual reality systems have been constantly playing a key role. For example, the Immersion Corporation's line of medical haptic-based systems ranges from surgical simulations to training for intravenous catheterization. The Immersion Corporation also designs a series of touch sensitive haptic applications that can be applied within the interior of cars. The applications include touch screens for navigation and hands-free communication, as well as rotary devices that can be adapted to the original equipment manufacturer's designs and requirements. The range of applications for the Immersion Corporation's haptic devices also includes commercially available products, such as fitness equipment, touch screen telephones and self-service counters.

There exists extensive research and publications in the field of haptic-based virtual rehabilitation. Most of the research relied on preliminary evidence that supported the introduction of haptic technology to virtual rehabilitation. Proving this evidence came about through clinical trials, such as Heidi Sveistrup's work [32] which compared virtual reality-delivered exercise programs to conventional exercise programs for physical rehabilitation. The result of these clinical trials proved the effectiveness of VR-based haptic motor rehabilitative systems over conventional systems. It also helped to emphasize the benefits of using VR within rehabilitative systems, such as the ability to tailor exercises to suit the needs of patients, as well the ability to develop a standard

for analyzing and assessing the progress of patients [33]. Furthermore, the author delves into other research that had been carried out in terms of specific applications for upper and lower extremity motor function, balance and locomotion. Referring to Broeren et al., [34] an assessment was carried out for an application of haptics in a virtual rehabilitative system, where the PHANTOM haptic interface was used and the patients were required to perform simple tasks such as reaching and grasping. A similar application was also used in a more complex system, where a patient navigating through a labyrinth reveals more subtle aspects of the effect of his/her neurological disease by examining the amplitude of the tremors and their frequency, the control over the movements and the patient's speed [35].

In previous work at the DISCOVER Lab [17, 36], the initial version of a VR system was developed to include a set of exercises that relied on force feedback mechanism. These exercises were obtained by incorporating common tests that OT's have been using, such as the Jebsen Hand [37] and the Box and Block test [38]. The system was then subjected to analysis by a group of five occupational therapists from the Ottawa General Hospital Rehabilitation Center aged between 20s and 60s, who volunteered their time to assess the effectiveness of the VR system, and the implemented exercises. The aim of the therapists' assessments was to collect as much feedback for the VR system as possible, such that all aspects of the system, including the hardware, software and graphics, were properly scrutinized. The result of the evaluation was a new set of exercises that better represented common daily activities that would appeal more to patients, as well as an the automated progress measurement component of the system.

While the use of haptics for rehabilitation has a number of advantages, it also comes with certain shortcomings. For example, the factory pre-installed calibration method for the CyberGlove system takes about 5~10 minutes to get a decent calibration for a healthy person. This will be even more problematic in the case of patients who have severe disabilities in their upper extremities, to the extent that their hand's motor functions are not physically suitable for use in the normal calibration process. In order to avoid having to put a patient through strenuous hand calibration exercises, a new

means of calibrating the CyberGlove hand had to be devised. In [12], a method was proposed that used a simple 2D digital camera to capture an image of the user's hand. The hand would be placed on a chess board background, and the image would be taken vertically. The image would then be used to determine the size of the user's hand, as well as the length of specific segments of the hand. The length of the segments would then be fed into an artificial neural network, which will determine the valid calibration data needed for this user, without having the user undergo any further calibration procedures. There are several reasons for using an artificial neural network to enhance the calibration process:

1. The use of artificial neural networks in calibrating complex systems has been thoroughly researched and tried
2. Artificial neural networks are significantly powerful mechanisms for identifying underlying relationships between various factors. This would be valuable since, during the calibration trials, a direct and constant relationship was identified between the size of the segments of the user's hand, and the actual corresponding sensor readings.
3. After developing the neural networks needed for each sensor, determining the suitable calibration data for each new user will become a very simple and fast process.

As we had already mentioned, the use of artificial neural networks in calibrating complex technical systems is not a novel approach, but has been attempted before by other researchers in the field. In [41], Griffin et al. modeled the opening and closing movements of a robot manipulator. The movements of the end gripper of the robot manipulator are controlled by the thumb and index finger of the user, where the distance between the tips of each finger is used. Their method consisted of recording the readings of the required sensors while the user places his or her thumb and index finger together and moves the other fingers for a period of about 40 seconds. They then build a nominal hand kinematic model using the recorded sensor data, and calculate the distances between the thumb and index fingertips. These distances are then considered as the errors to minimize, and the least squares method is then applied to produce a

model for each user. The method used by Griffen et al. was further extended to all five fingers of the hand in [42], for applications of virtual reality simulation for astronaut training and tele-medicine. Another example of the use of artificial neural networks for the calibration of a dataglove can be found in [43]. Fischer et al. built a 3D Cartesian model of a four-fingered robotic hand by using a non-linear learning calibration mechanism that uses a neural network. The method used to obtain the actual Cartesian positions of each fingertip of the hand, which will then be used in training the artificial neural network, consisted of marking each of the four fingertips with a colored pin, and using two cameras to capture images of the fingertips. Although the method can achieve the fingertip positions with an error of less than 1.8mm, a number of limitations exist:

1. Time synchronization between the dataglove's data and the visual data captured by the two cameras is essential.
2. The method requires the use of two calibrated, color cameras.
3. The method produces the Cartesian positions of the fingertips only.

Another approach can be found in [44], where Kahlesz et al. take into account the cross-coupling effects of the flex sensors of the metacarpal (MCP) joints on the abduction angles in between these joints. The proposed method creates an iso-surface for each abduction sensor, with one for each trajectory:

1. Only the flex sensor to the left of the in-between abduction angle is bent.
2. Only the flex sensor to the right of the in-between abduction angle is bent.
3. Both the right and the left flex sensors are bent.

These surface values are then stored in a hash table, allowing for fast look-up of the abduction angle. Furthermore, B. Wang et al., in [18], have modified the work to consider the effects of the adjacent flex sensors to be linear, producing a modified means of calibrating the sensors. The user must move the neighboring metacarpal (MCP) joints both independently and jointly, after which the least squares method is applied to produce the cross parameters for the flexion sensors.

In another related research [45], the relationship between the human hand and the human body was taken into consideration as well. This was achieved by looking into the body distance between the segments and the joints. The Hidden Markov Model was also used to recognize complete motion and sequence of motions that are common in human behavior instead of modeling each static position in time. This will introduce much more stability in the motion and can be used to enhance the calibration of the model. However, in this method, minor twitches and finger gestures will be masked by this calibration model which could be advantageous in human to haptic motion translation, but not in the field of physical rehabilitation. For virtual rehabilitation applications, the minor twitches and finger gestures are indicative of the motor issues that a patient is suffering from and would be essential to capture and monitor. Thus, it is very important not to mask any abnormal motion as it would be significant in the diagnosis and treatment of a patient.

Other applications of neural networks for calibrating systems include [46], where a neural network was used to reduce the non-linear effects of an imaging system based on charge-coupled cameras (CCD), where the artificial neural network would learn the mappings between the actual distorted image points and their corresponding pinhole camera image points. Another application consists of using a neural network for recreating 3D coordinates of an object, using the 2D coordinates obtained from two images using a two camera system for measuring these coordinates. The details of the method can be found in [47]. In [48], Y. Liu calibrates an industrial prototype microwave six-port device, which is commonly used in many industrial online microwave measurement instruments, using an artificial neural network, achieving high accuracy and minimizing errors over a wide dynamic range of measurements. J. M. Dias Pereira et al. [49] also propose the use of an artificial neural network to counteract the effects of temperature drift errors on small amplitude signals from transducers that could affect the accuracy of the results from measurement systems.

# Chapter 3

# The Proposed System

The system developed consists of an underlying framework that manages the integration of the haptics device with the virtual environment, and renders the 3D graphics into the virtual environment as well. The system also consists of a user interface that facilitates the user's interaction with the system. It allows Occupational Therapists to setup the system and configure the exercises as the patients use the system. It also allows the therapists to analyze the results of each exercise. The Progress Measurement component can also perform an analysis of the patient's results and determine if the patient needs to advance to a higher level or not. The main components are the actual exercises, represented by the 3D graphics. To better understand the structure of the proposed system, Figure 2 presents the different components of the system.

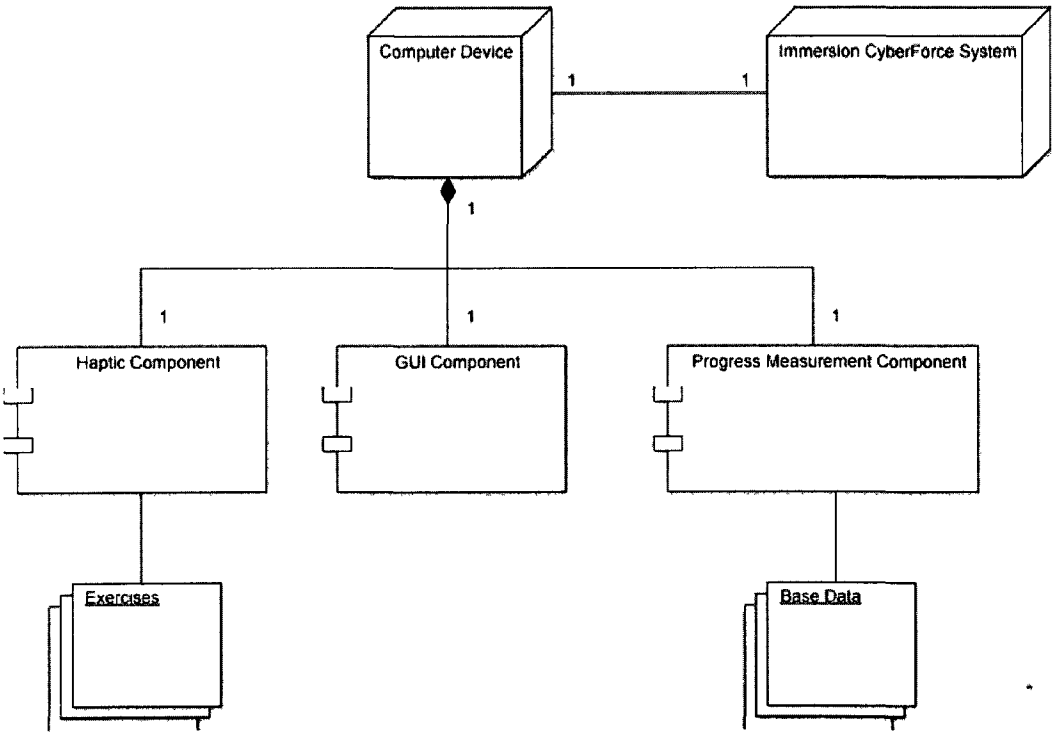


Figure 2: Deployment diagram of the proposed system

The haptic component of the system is used to generate the exercises prior to using the system. The exercises would be included within the system, when deployed in rehabilitation centres. As a part of the deployment procedure, the Occupational Therapists would also be required to define the characteristics of the volunteers needed for defining the base data to be used as the input for the Progress Measurement component. Once the patients begin using the system, each run of an exercise will produce a set of results, and these results can be fed into the Progress Measurement component for further analysis.

After interviewing several occupational therapists and gathering their feedback [50], new exercises were created that provide the patients with more realistic exercises, inspired by common daily activities. These exercises include the following scenarios:

- Tea pouring and drinking
- Moving common kitchen items onto shelves
- Drinking soup from a bowl with a spoon

More exercises are expected in the future, which will also be taken out of the context of daily activities. We will discuss these exercises shortly, but lets us describe the system's functionality by looking at its user interface first.

## 3.1. User Interface

The interface allows the OT to interact with the system in a simple and user-friendly manner. It provides the OT with the ability to setup the base data, against which patient data is correlated, and to configure each exercise in accordance with a patient's progress. The following sequence of screenshots represents the user interface, as it would be used by Occupational Therapists, starting with Figure 3 below:

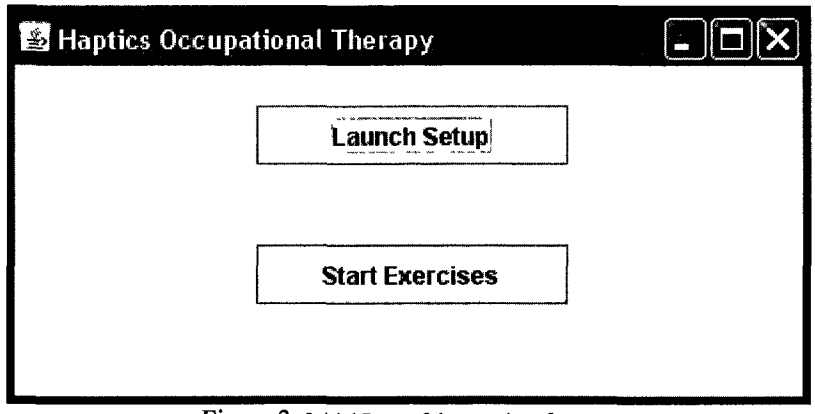


Figure 3: Initial Page of the user interface.

An OT must select to either begin the system setup, which includes defining the base data for correlation, or launching the exercise page.

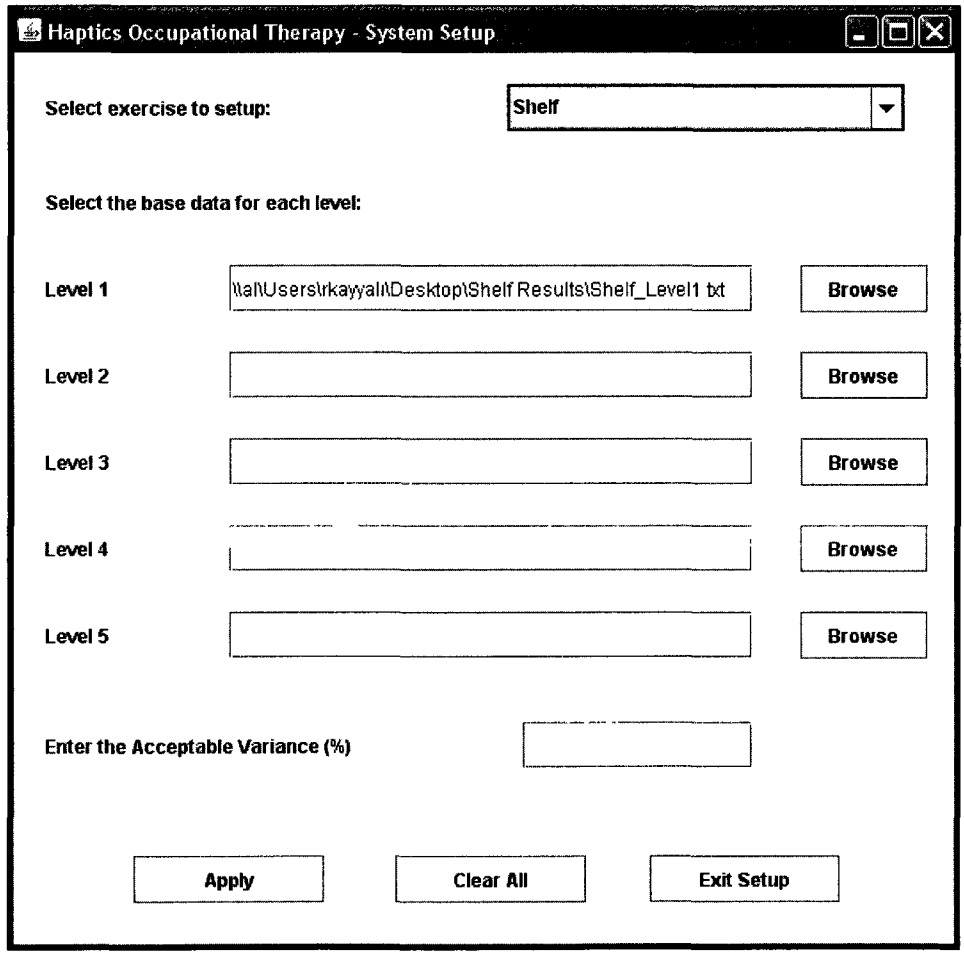


Figure 4: System Setup Page.

The OT must select each exercise to setup (Figure 4), from the drop-down list, and browse to the data file for each level. The OT would then enter the acceptable degree of variance between the patient's data and the base data.

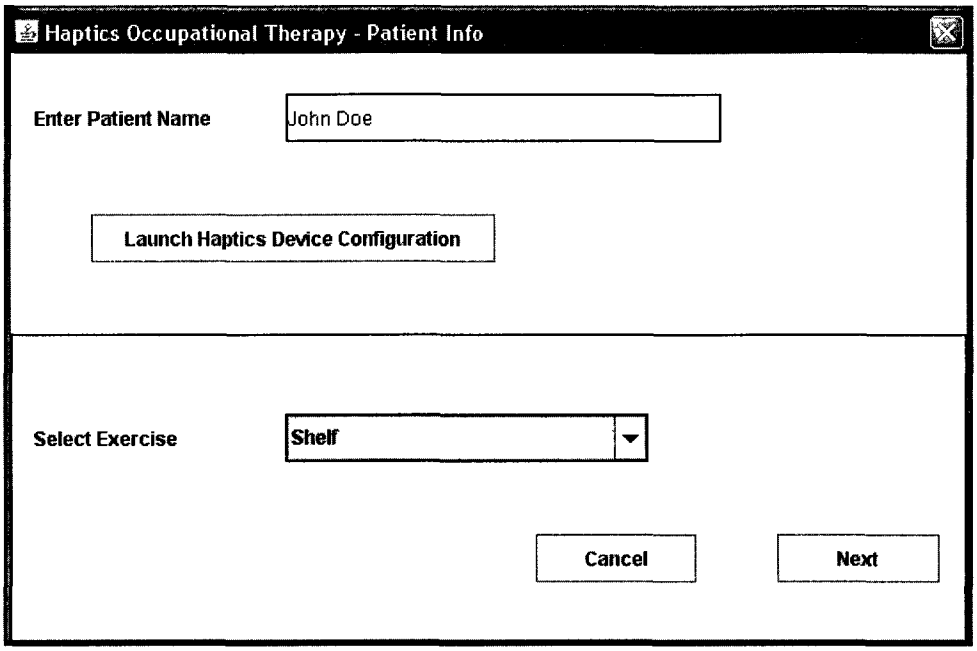


Figure 5: Exercise Page.

The OT enters the patient's name (Figure 5), against which a folder will be created with the patient's name and current date and time. The OT would proceed to configure the haptic device for the patient using the device-specific application. This step would initially require some time to configure the device, but the configuration can be saved into a file and loaded each time.

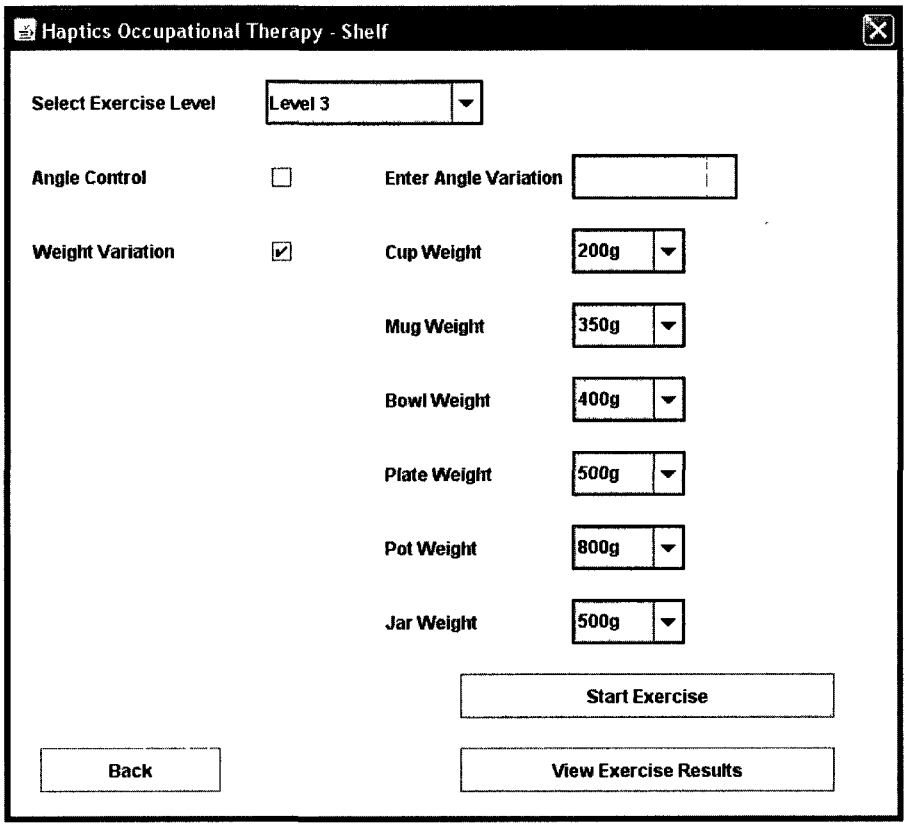


Figure 6: Exercise Configuration Page.

The OT then selects the level for the exercise, as shown in Figure 6. This provides pre-populated weight variations for certain objects within the exercise. The exercise is then launched, with the required configurations. Once the patient is done, the OT can review the results and analyze them.

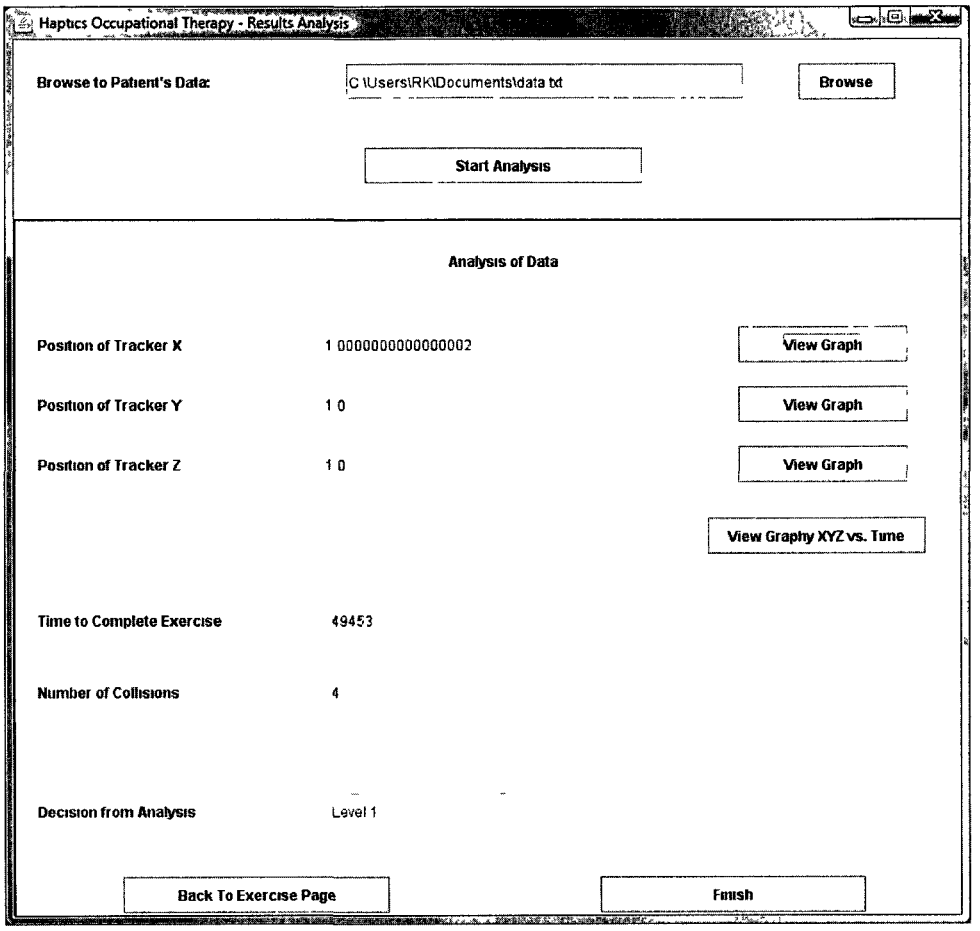


Figure 7: Result Analysis Page.

For the analysis, the OT can choose the data against which to compare against. This allows the OT to perform the analysis at a later time. The results are displayed in the lower portion of the page. The correlation of each of positions of each of the Tracker X, Tracker Y and Tracker Z are displayed (Figure 7). The OT can also view a graph populated of the respective tracker's positions against time (Figure 8). The page also displays the final decision of the analysis, which is a determination of which level the patient must proceed to.

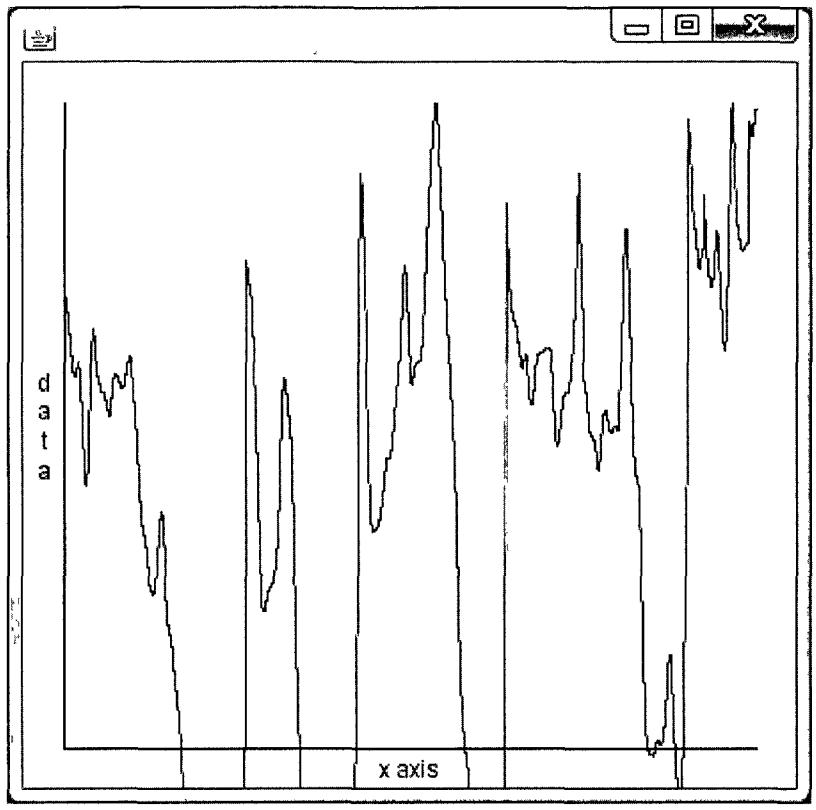


Figure 8: Graph of Tracker X positions vs. Time.

## 3.2. Rehab Exercises

### 3.2.1. Maze Exercise

As mentioned earlier in this section, the system offers a number of exercises. The first one is the 3D maze. The 3D Maze exercise essentially consists of a cylindrically shaped object that would be used to traverse a path within a maze from the starting point until the end point. The maze itself is a simple 3D structure, built using mesh cube objects. The following requirements, which were used to tailor exercises, were requested by the OT's:

- Ability to vary the angle of the maze from 45° to -45° the horizontal axes.
- Several levels of complexity.
- Ability to track the speed of patients as they move the stick.
- Ability to alter the weight of the stick being moved through the maze.

The main objective of the 3D-maze exercise is to increase the motion range of patients, along the horizontal and vertical planes. This exercise also allows patients to increase hand's steadiness and eye-to-hand synchronization. The number of collisions between the stick and the maze walls is computed. Speed tracking is another important requirement in measuring a patient's performance. The haptic devices currently employed in the VR system do not provide a means of measuring the speed of the hand as it moves. We addressed this issue by using a spherical ball. The patient starts the exercise but, after a specified time delay, the ball begins to traverse the path of the maze. Once the ball collides with the stick, this means that the patient is moving too slowly and the OT can determine if a longer time delay is needed. Figure 9 shows level 4 of the Maze exercise with Speed Tracking turned off.

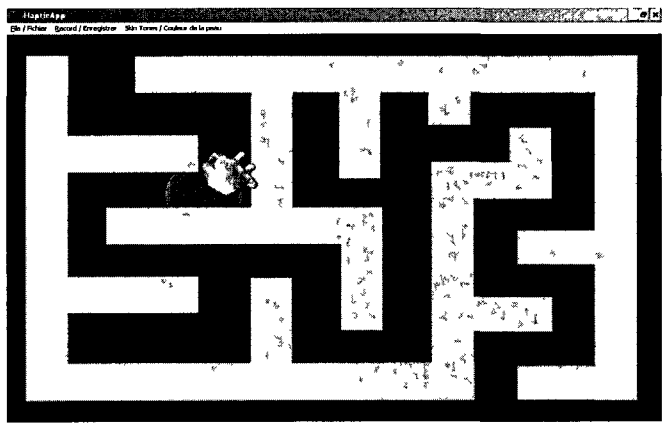

Figure 9: Maze Exercise Level 4

### 3.2.2. Shelf Exercise

Another exercise if the Shelf Exercise. This essentially consists of two shelves, fixed in place within the virtual environment, and a set of common items that can be found in all kitchens. The purpose of the exercise is to have the patient pick up the items, placed alongside the shelves, and place them on the shelves. The items themselves range in size and weight, and are made distinguishable through the use of appropriate colors. A sample of the exercise can be seen in Figure 10. The exercise helps patients practice the action of gripping items with their hands, and the motion of

carrying items from a lower level to a higher level, as well as forward and backward movements of the arm. The variations in the sizes and shapes of the items also require different hand grips of the items.

Figure 10: Shelf Exercise

### 3.2.3. Tea Exercise

The Tea exercise includes several levels of difficulty, with each advanced level including a heavier tea pot. The exercise also allows the patient to lift the teacup, separate of the saucer. This also helps the patient learn to perform more delicate tasks which include the use of the thumb and index finger only. The patient will need to hold the teacup in a steady grasp to prevent the tea from overflowing, as well as grasping the tea pot from a smaller handle while balancing the weight of the tea pot. There is also a variation in the angle of the wrist required to hold the tea pot and the teacup. The following figure, Figure 11, provides a sample of the exercise.

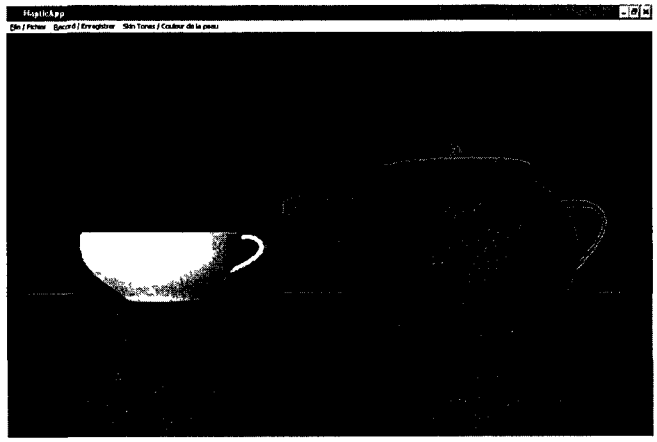

Figure 11: Tea Exercise

### 3.2.4. Soup Exercise

The Soup exercise consists of a bowl of soup and a spoon, which the patient lifts and uses to drink the soup. Again, this sort of exercises allows the patients to practice holding small objects in their hands, and learn to move their arms over a more measured and precise path, to their mouths. This exercise would potentially be difficult to perform by a patient early in the rehabilitation process. It would require a lot of control over the fingers, hand and arm, which would be possible during the more advanced levels of rehabilitation. A sample of the exercise can be seen in Figure 12.

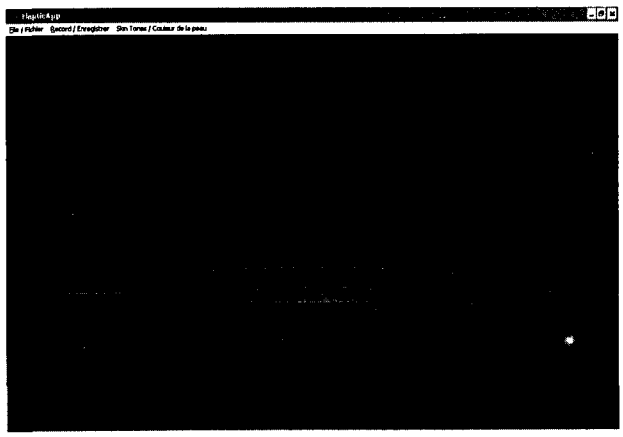

Figure 12: Soup Exercise

## 3.3. Progress Measurement

The progress measurement component of the system provides the Occupational Therapist with an automated means for monitoring the progress of the patient. The framework of the system gathers and records certain data during an exercise. The data collected is:

- Positions of the X Tracker
- Positions of the Y Tracker
- Positions of the Z Tracker
- Thumb's Proximal in radians
- Index's Proximal in radians
- Middle finger's Proximal in radians
- Ring finger's Proximal in radians
- Pinky finger's Proximal in radians
- Collisions between two or more objects in the virtual environment
- Time, in milliseconds

These values are used to track the motion of the hand within the virtual environment. They also show the strength with which the hand is gripping objects and the angles of the fingers with respect to their knuckles. This indicates whether or not the patient's grip is improving, and if the patient is gaining more strength and control over his/her fingers and arm.

### 3.3.1. Design

In order to achieve a complete analysis against all types of benchmarks, we need to gather data for each possible type of healthy users, using each possible exercise at each possible level. With four exercises, five levels and five possible types of users, we have a hundred combinations to gather data against. Potentially, the number of user types can increase to match the new characteristics of the end users. This will drive the number of combinations of exercises, levels and user types even higher. However, this will

definitely be the case as more exercises are created, and more levels are designed. In order to avoid this unmanageable exponential growth of combinations, we needed to curtail the overload of possible base data. After examining the data that was collected over the trial period, we have made the following set of lax conclusions:

- As an end user begins using the system, he/she will be set to level 1. This is indicative of the lack of haptic experience that the end user would have, as it is comparable with the volunteer carrying no computer skills (i.e. volunteer 1).
- As the end user advances over time, he/she will become more of an expert in using the haptics device. This will lead to the end user becoming more and more comparable with the skills of a haptics expert, hence volunteer 5.
- These sets of data are only applicable in the scenario where the system is deployed in a "healthy" environment, such as a research lab. When the system is installed in a rehabilitation center, the base data will have to be collected from volunteers with different characteristics. These characteristics will be determined at a later stage with the help and knowledge of professional occupational therapists.

Following from these conclusions, we took the decision to simply represent each level by a volunteer with certain characteristics, which we have previously defined in the trial period.

### 3.3.2. Implementation

The progress measurement component analyses the data that was collected during an exercise and correlates it with the base data that was provided during the system setup. The base data used in the system was collected during the trial phase of the system. The data collected from the fifth trial of each exercise, for each volunteer, was used for the benchmark data. This was decided as the first couple of trials provided unreliable data since most of the volunteers were not yet accustomed to the system. Each volunteer represents a level in each exercise, with volunteer 1 setting the basis for level 1, volunteer 2 representing level 2, and so on. During the setup phase, these data

files are parsed and the required values extracted. The values are then stored in a custom designed data structure that allows for faster runtime analysis.

Analysis of the data is then carried out by the Occupational Therapist at will. The OT simply has to select the data file of the patient to analyze. The correlation is performed based on Pearson's Correlation Coefficient. The formula used is as follows:

$$r = \frac{n\sum x_i^2 y_i^2 - \sum x_i y_i}{\sqrt{n\sum x_i^2 - \left(\sum x_i\right)^2}\sqrt{n\sum y_i^2 - \left(\sum y_i\right)^2}}$$

where $x$ is an element of the base data and $y$ is an element of the patient's data. This provides the coefficient of the correlation between the patient and the base data. The calculation is carried out for every exercise level's set of base data, until an acceptable result is obtained. To determine if a result is acceptable, the following characteristics of the result must be obtained:

- The result is positive in value. This indicates that the two sets of values have a positive correlation. The significance of the positive correlation lies in the linear relationship with the two sets of correlated values, where if one set of values tends to increase, the other set of values also tends to increase. This shows that the values are similar in nature.
- If the result is positive in value, the variance percentage is applied, such that the final decision is based on:

  $$Correlation\ Result \pm (Variance / 100)$$

  This variance value is what the occupational therapist declares during the setup of the system.
- The correlation result is considered acceptable when the correlation value is between 0.7 and 1.0. We chose to make the correlation a strong positive one as this would guarantee that the patient is strong enough to move onto the next level. Relying on a weak correlation would indicate that the patient could still benefit more from repeating the exercise for a while longer.

At this point, the level for the patient to proceed to next becomes the level for which this acceptable result is obtained. This might the same level the patient is on, a lower level or a higher one, depending on how the patient is improving or maybe even getting worse.

# Chapter 4

# Performance Evaluation

## 4.1. Rehab Exercises

In order to determine the effectiveness of the Haptic Virtual Rehabilitation system, trials have been carried out with five volunteers. The volunteers were selected with the following characteristics:

- Volunteer 1: Minimal computer skills
- Volunteer 2: Recently undergone a laparoscopic surgery for shoulder re-alignment
- Volunteer 3: No prior knowledge of the Immersion CyberForce system
- Volunteer 4: Some prior experience with the Immersion CyberForce system
- Volunteer 5: A Haptics expert

The results of these trials were captured and analyzed. The data provided us with a means of following the volunteers' improvement and allowed us to gather feedback from users of the system. The data gathered also proved useful in developing the progress measurement component, when used as a benchmark against future patients' data.

### 4.1.1. Trial Procedure

Each of the volunteers went through the same set of steps during the trials. These steps included, wearing the Immersion CyberForce system, configuring the hardware, performing the exercises in the following order: Soup exercise, Tea exercise with 1N (Newton) weight, Tea exercise with 15N weight, Shelf exercise. For each exercise, the required values were gathered in a log file and saved. The same process was carried out five times over a period of three days.

### 4.1.2. Results

The trial results were gathered by keeping logs of the haptic data received over the system's framework. The data gathered included:

- Tracker's X, Y and Z coordinates, with respect to each volunteer's configured axes of reference,
- Angles of the five fingers on the right hand with respect to the palm of the hand such that the angle is taken from the joint between the fingers and the palm
- Number of collisions with other objects in the virtual environment
- Time, in milliseconds, for the logged values.

The results for the Soup exercise, as obtained from Volunteer 1, can be seen in Figures 13 and 14, with Figure 13 representing the Tracker Positions (X, Y and Z coordinates) for the first trial, and Figure 14 representing the same data types for the fifth trial. One can see that the values are much more consistent in the last trial, and the range of motion is more stable. This can be determined from the oscillations in position, which appear to be smaller and more consistent in trial 5. Such results can be used by occupational therapists to measure the stability of a patient's arm, and the improvement in controlling the arm and its ability to move along the required path of motion. Similar result patterns were observed for all volunteers, with the less experienced volunteers showing a slightly less significant improvement than those with previous haptic experience. The significance of the oscillation pattern for the results of the Soup exercise is in the arm and hand movement required to perform the exercise. The aim of the exercise requires the user to grab the spoon, and then move the spoon to and from the soup bowl, towards his or her mouth. These actions indicate an almost constant change in the X, Y and Z coordinates of the Position Tracker. This means that the more consistent the oscillation pattern of the results is, the more control the user is having over his or her actions.

Other test results have shown more forms of improvement with regards to a volunteer's performance. For example, the task completion time continued to improve, for all volunteers, over the course of the trial period. This can be seen in Figure 15. Volunteer 1 took the longest time to complete the Tea Exercise, which was carried out using a 1N weight for the teapot. This was due to the volunteer's difficulty in trying to visualize the virtual environment as a 3D environment. However, the improvements in time to complete each exercise can be attributed to the volunteers becoming more accustomed to the exercises, instead of actual improvements in their performance abilities. This is the case with the use of healthy volunteers that do not have problems with the upper extremities. In order to fully establish the effectiveness of the system, actual patients would have to be used in future trials.

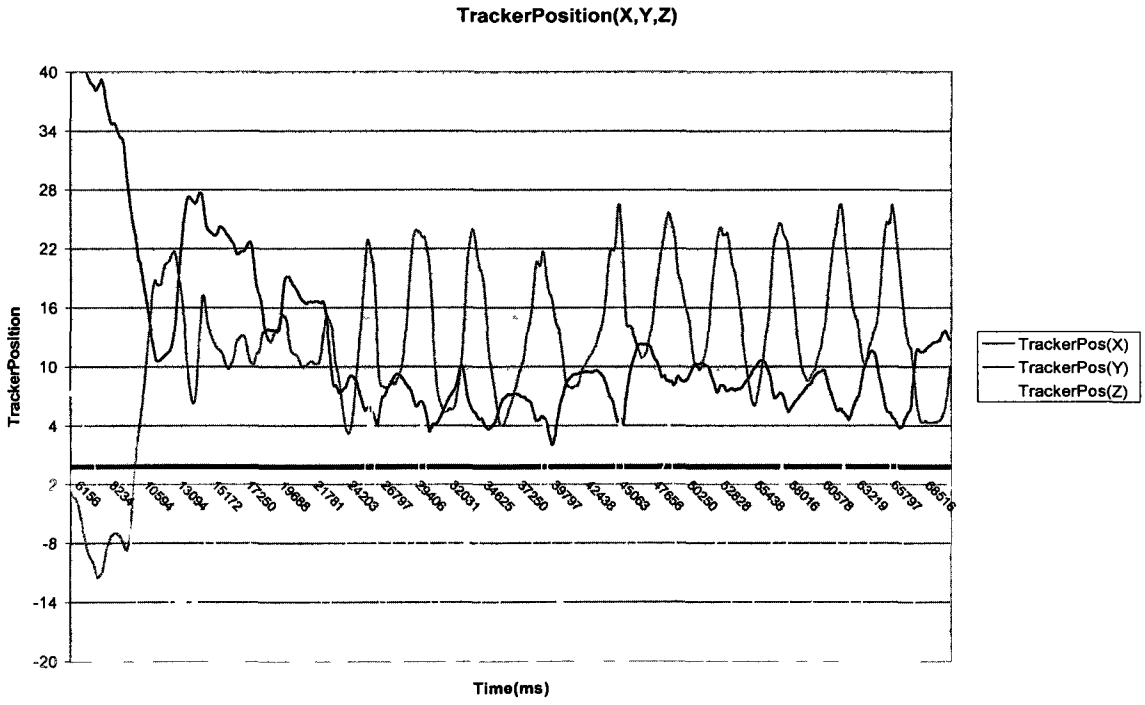


Figure 13: Soup Exercise Tracker Positions for X, Y, Z Coordinates Trial 1

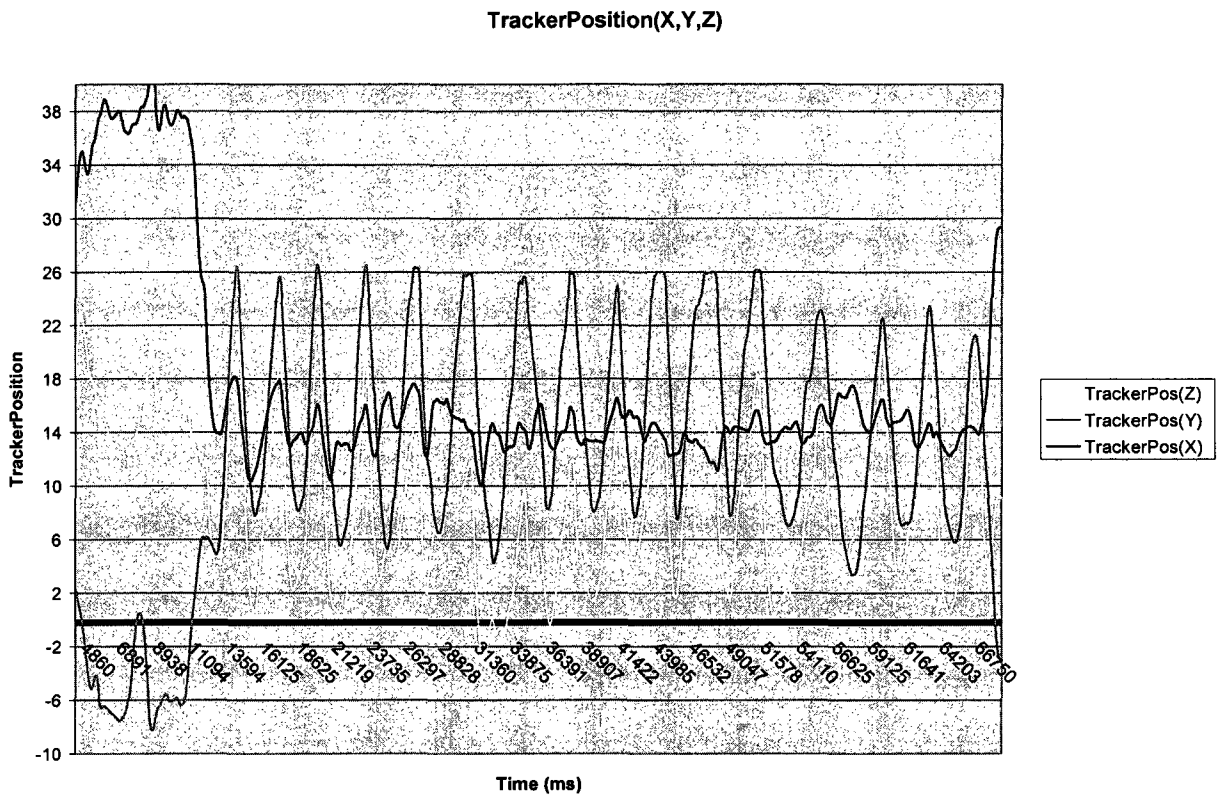


Figure 14: Soup Exercise Tracker Positions for X, Y, Z Coordinates Trial 5

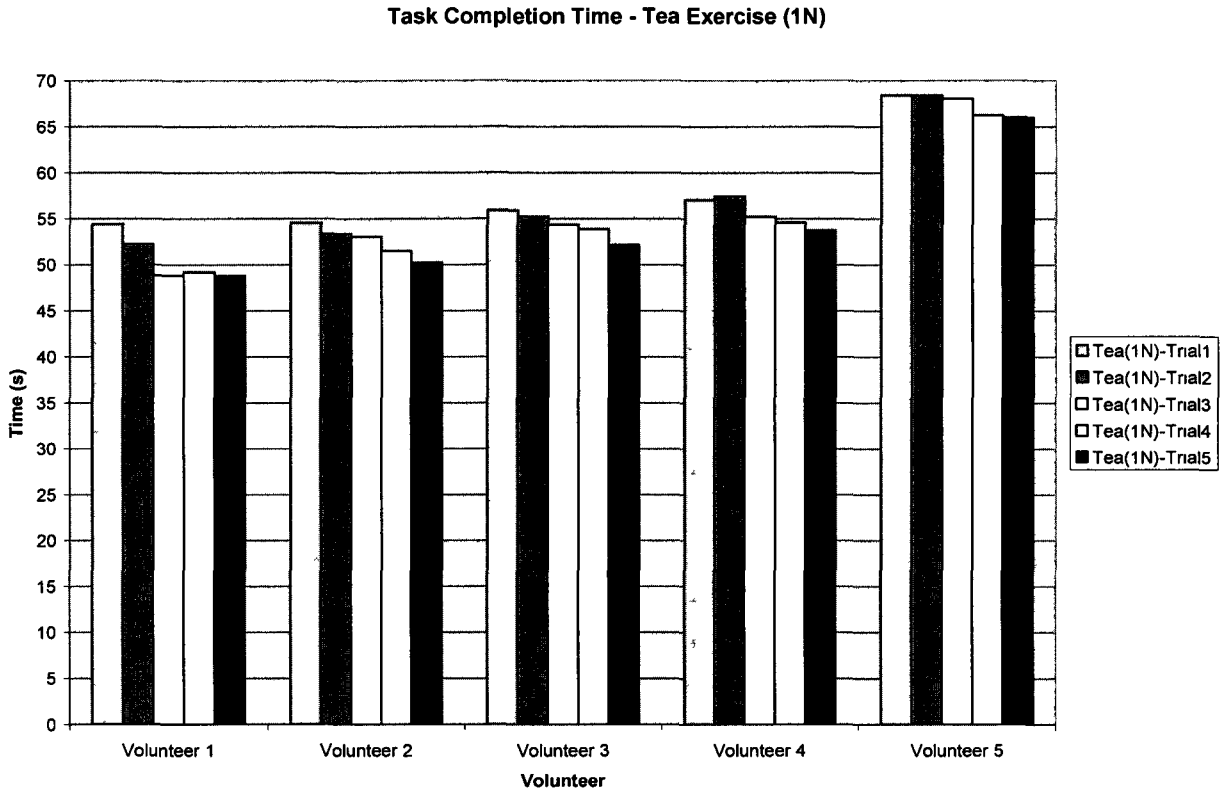


Figure 15: Tea Exercise (1N) Task Completion Time

### 4.1.3. Feedback

Of the feedback received from the volunteers, the most significant issues were related to the hardware used in the system. Most of the volunteers placed emphasis on their inability to view their actions being reflected properly in the virtual environment. This is mainly due to the configuration process and the fact that the CyberForce does not accommodate for different hand sizes. Another problem with the hardware is the restricted workspace available, where the limits are determined by the length of the cables and wires, and by the CyberArm. The workspace limitations caused some of the volunteers to fail at performing the Shelf exercise, and re-configuration of the Immersion CyberForce system was required to adjust the center of the workspace.

Further issues expressed by some of the volunteers, was their inability to realize the screen rendering of the exercise as a 3D environment. This issue was especially

expressed by the volunteers that had little or no prior experience with haptic devices or virtual environments.

On the other hand, the volunteers had also expressed their preference for the exercises based on the more familiar daily-life activities. The majority of the volunteers found it easy to understand the aim of each exercise. The choice of colors used in the rendering of the virtual environments was also admired by most of the volunteers, as the colors were not too bright but were different enough for them to differentiate the objects. Some volunteers, specifically Volunteers 4 and 5, had expressed preferences for improvements in the quality of the graphics, but the other volunteers did not complain. This issue will probably be taken up as a future improvement to the system.

## 4.2. Progress Measurement

In order to properly test the progress measurement component, we had to ensure that the data analysis section worked as expected. Hence, passing in an identical set of data for the patient's results, and setting up the base data also with identical data sets, meant that the correlation results for all of the tracker positions, X, Y and Z, and for the five finger proximals, must all be a value of 1.0. This indicates that the correlation coefficient is 1.0 which is the expected result when the two items being correlated are identical. It would also mean that the expected level would be the minimum level possible that returns the acceptable correlation coefficient; in this case it would be level 1.

After running the test several times with several different data sets, each set returned the exact correlation coefficient expected, 1.0, and the final decision level was "level 1". This helped to prove that the data analysis algorithm functioned as designed.

The next set of tests was designed to show if the progress measurement system would be able to recognize improvements in the user's performance. In order to do this, each of the volunteers would have to undergo several trials of the same chosen exercise, in this case the maze exercise. In each trial, the volunteer will naturally improve as he

or she becomes more accustomed to the exercise. Since the progress measurement component was designed for only 5 levels of improvement, we chose to work with volunteers 1 to 4 only. The decision to exclude the fifth volunteer allowed the trials to go on without having to alter the progress measurement component to include a sixth level of improvement.

The trial begins by each volunteer performing the maze exercise only once. After each run, the volunteer's results would be analyzed by the progress analysis component and the resulting level is noted. The time taken to complete each exercise, as well as the number of collisions, was also noted. The same process was repeated for another five times, and each time the result of the progress analysis component was noted. As was expected, the results obtained for the volunteers showed an improvement after the fifth trial of the maze exercise. In order to verify the results of the progress analysis component, the times taken to complete each exercises run were also compared, as well as the number of collisions that occurred in each run. The following table, Table 1, details how the volunteers performed:

| | Trial 1 | Trial 2 | Trial 3 | Trial 4 | Trial 5 |
|---|---|---|---|---|---|
| Volunteer 1 | Level 1 | Level 1 | Level 1 | Level 1 | Level 1 |
| Volunteer 2 | Level 2 | Level 2 | Level 3 | Level 3 | Level 3 |
| Volunteer 3 | Level 3 | Level 3 | Level 3 | Level 4 | Level 4 |
| Volunteer 4 | Level 4 | Level 4 | Level 4 | Level 5 | Level 5 |

Table 1: Results of Performance Measurement Analysis for Volunteers 1 - 4 during Trials 1 – 5

Volunteers 2, 3 and 4 all showed advances in their exercise levels, but volunteer 1 was yet determined to be at the same level. However, an analysis of the time it took volunteer 1 to complete the fifth trial was significantly shorter than it took to finish the first trial, as can be seen in Figure 16.

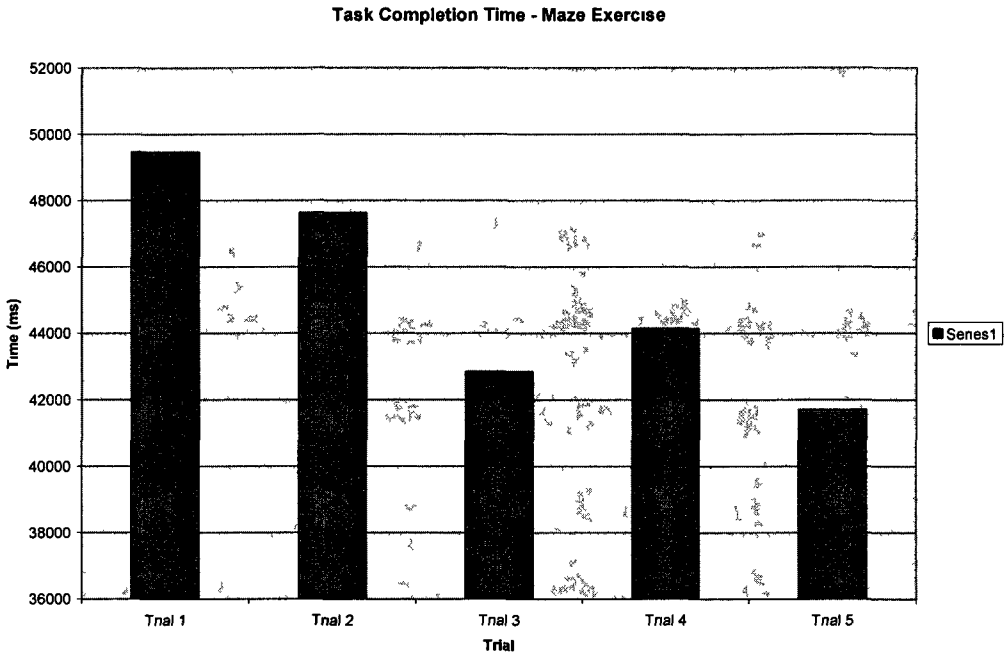


Figure 16 Chart showing time-to-completion progress for Volunteer 1

# Chapter 5

# CyberGlove Calibration Method

During the development of the system and the various trials that were carried out, there was always the issue of the calibration process that was deemed necessary to overcome. The process itself involves the user wearing the CyberGlove and the CyberGrasp systems and performing certain gestures, as required by the Immersion CyberForce software. The use of the calibration system is a necessary step to allow for accurate rendering of the user's hand within the virtual environment, and the effective force feedback on the user's hand. However, for patients with severe deformities of their upper extremities, the calibration process will prove to be an almost impossible task. The gestures require control over the hand and the fingers, such that the required calibration values of the sensor angles can be measured, which some patients may not be able to endure. Furthermore, the actual calibration process itself takes between 5 to 10 minutes to complete, when performed on a healthy person.

The above mentioned hindrances meant that a new calibration mechanism had to be designed. Several possible designs were suggested but the use of Artificial Neural Networks (ANN) was deemed the most appropriate. An Artificial Neural Networks is either a mathematical or a computational model, which tries to simulate biological neural networks, through symmetry with their structure or functional aspects. It is also an adaptive system that can change its structure or function during the learning phase, based on internal or external information that is input into the network. Neural networks can be used to find relationships or patterns in data or to model complex relationships between inputs and outputs [51]. There are a few advantages to using ANN's, such as:

1. The successful applications of ANN's in complex calibration systems
2. The ability of ANN's to determine underlying patterns or relationships between different values within a large system.

3. Once each sensor of the CyberGlove is matched to its most suitable ANN, the ability to generate the new calibration data for new patients will be fast and simple.

To explain the above-mentioned points further, several examples of the applications of neural networks can be analyzed. In D. Choi et al. [46], a neural network was used in a technique to diminish the nonlinear distortions in images produced by a charge coupled camera device. The neural network would learn to map between the distorted image points and the corresponding pinhole camera image points. This technique was re-usable for all filtering algorithms that assumed a pinhole model for the camera imaging system. Another example on the use of neural networks is detailed in [47]. The study included performing a statistical analysis on two techniques for building three dimensional coordinates of objects from pairs of two dimensional image coordinates, with the first technique using linear transformations of the coordinate pairs, while the second used artificial neural networks. A third example can be found in [48], where a microwave six-port device was calibrated using an artificial neural network technique for nonlinear approximations. Furthermore, while collecting calibration data, a direct and consistent relationship between the human hand's segments sizes and their corresponding sensor readings was beginning to emerge. With the use of a neural network, such a relationship or pattern could be determined easily.

The calibration process of the CyberForce system required measurements for all three components of the system. However, the use of the ANN is to be applied to the values that are gathered using the CyberGlove only. As was previously mentioned, the CyberGlove has 22 sensors for capturing the following measurements:

- finger flexing
- abduction
- thumb roll
- palm arch
- wrist yaw

- pitch

Each sensor produces an 8-bit unsigned integer output that is in the range of 0-255, which represents a sub-degree resolution of the joint movement. The joint angle can then be calculated as

$$\theta = g(r - x), \qquad (1)$$

where $x$ is the sensor's raw data reading, $g$ is the sensor gain and $r$ is the offset. This reduces the purposes of the calibration process into resolving the values of the gain and offset for each user, such that the user's hand gestures can be accurately rendered in the virtual environment.

The implementation of the neural network component in the VR system begins with the use of a camera and a chessboard. The camera is used to take a two dimensional image of the user's hand. These two dimensional images are then used to extract the lengths of the segments of the user's hand, as detailed in Figure 17. These lengths are then inputted into the neural network, and the outputs are the data collected from the manual calibration process for the same user. Finally, each sensor will be matched with the most suitable neural network, after a controlled training phase of the networks, where the selection will be based on the minimum mean square error when comparing the network's output with the actual manual calibration's data.

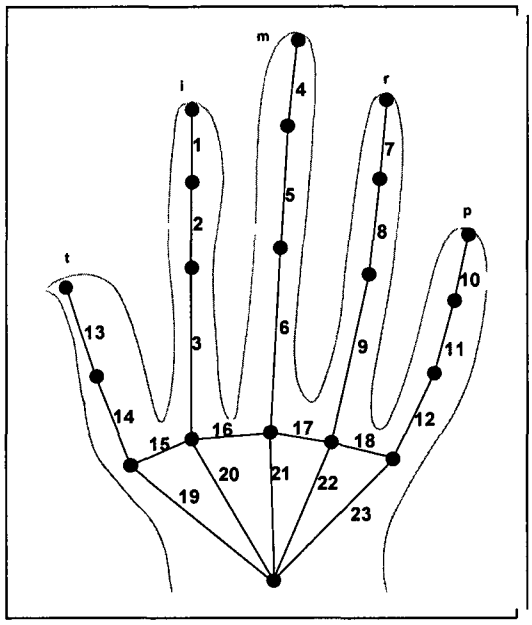


Figure 17: Indexed Hand Segments

However, since the training of the neural network relies on the use of only a few input sets of data, it becomes more difficult to find the best suited gain and offset values for the sensors. For this purpose, the system was enhanced to use the Least Square Regression method for estimating the parameters [18]. The Least Square Regression method is used to approximate the solution of over determined systems, where the number of equations exceeds the unknown variables [19].

For the purpose of this ANN, the Linear Least Squares method was used. A Linear Least Square Regression model implies that there is a linear combination or relationship between the different parameters [52]. The parameters under study here are the joints' angles and the actual readings from the corresponding sensors. As was detailed in [18], the human hand was modeled as is shown in Figure 18 [41]. The figure shows the various joints in the human hand and the different sensors reading off the hand within the CyberGlove. The model shows the human hand as having 31 Degrees of Freedom (DoF), where the DoF of each joint is the set of independent displacements that define the position and orientation of the joint [53]. According to the model chosen, each finger has four degrees of freedom,

1. One distal interchangeably (DIP) joint DoF
2. One proximal interphalangeal (PIP) joint DoF
3. Two metacarpophalangeal (MCP) joint DoF's

the thumb has five,

1. One interphalangeal (IP) joint DoF
2. One metacarpophalangeal (MCP) joint DoF
3. Three trapeziometacarpal (TM) joint DoF's

and the palm has four DoF's located at the base of each finger's carpometacarpals (CMC). Since each finger has 2 MCP DoF's, a cross-coupling effect will arise on the abduction angles in between the fingers. In [44] Kahlesz et al. have devised a method to counter-act this cross-coupling effect, which creates an iso-surface from the results of three experimental trajectories. However, this method produces a contradiction with the aim of trying to achieve the highest accuracy for the joint angles when compared between the rendering in the virtual environment and the user's hand. Yet, when analyzing the method used by B. Wang et al. in [45], the effect of the neighboring flex joints becomes linear, and the equation derived from the calibration process (1) becomes:

$$\theta_{abd} = g_{abd}(r_{abd} - x_{abd}) + (k_l x_l + k_r x_r + b), \qquad (2)$$

where $x_l$ and $x_r$ are adjacent MCP flexion sensor readings, $k_l$ , $k_r$ and $b$ are the cross parameters for flexion sensors. The results of the manual calibration process are then subjected to the Linear Square Regression method and the cross parameters are produced.

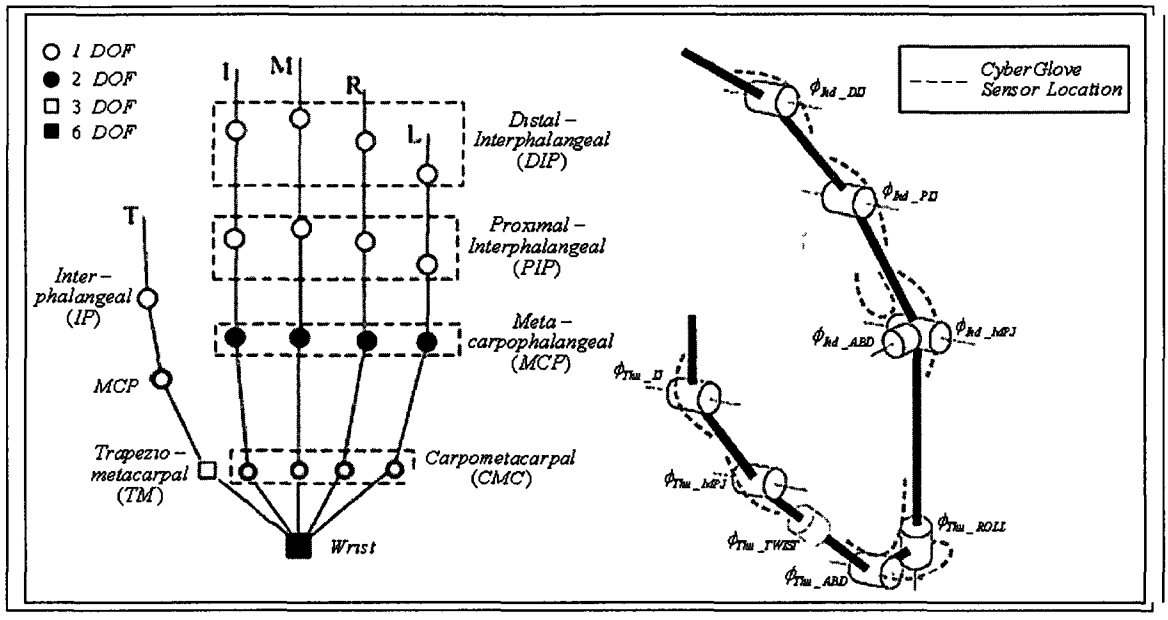


Figure 18: Model of human hand used

The new calibration process is needed to correct the calibration errors in the Virtual Hand Device Calibration Utility. Figures 19 – 20 emphasize the problems with the calibration software. The user is performing the action shown in Figure 20(a – b), where the user is bending the middle finger inwards towards the palm. The rendering of this action shows the finger as being held at a different angle. This is due to the method of calculating the abduction angles of the middle finger, where the calibration process is considers the raw sensor value of the abduction sensor only, while disregarding any effects of the adjacent flexion sensors. The adjacent flexion sensors should assist in determining the direction of the movement of the abduction angle, while the raw sensor value of the abduction sensor reflects the degree of the angle. In Figure 19, the rendering is showing the abduction angle as being of a considerate value, while Figure 20(a – b) shows that there is no angle between the Middle Finger and the Index Finger. This is due to the abduction sensor twisting which produces a raw sensor value that the Virtual Hand Device Calibration Utility translates into an actual abduction angle.

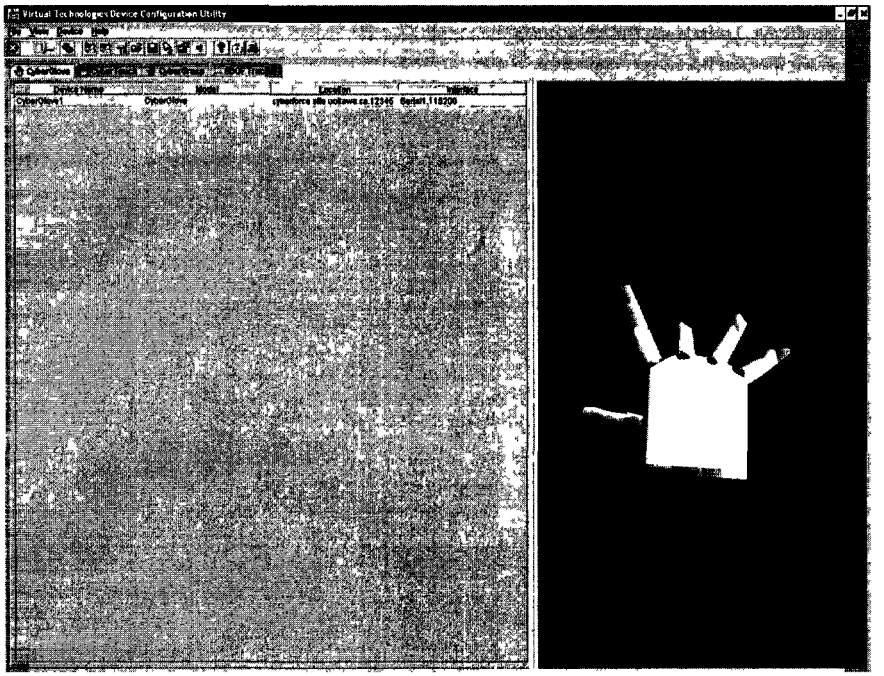

Figure 19: Screenshot of Virtual Hand Device Calibration Utility showing rendering of a hand with middle finger bent inwards

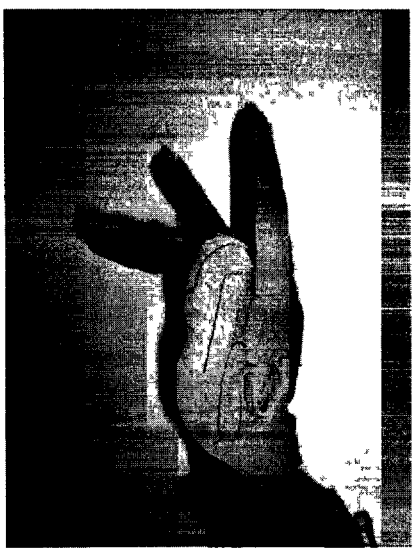

Figure 20(a): Actual hand performing gesture of bending middle finger inwards – left lateral view

(b): Actual hand performing gesture of bending middle finger inwards – dorsal view

## 5.1. Implementation of the LSR Analyzer

Since the factory-delivered calibration mechanism of the CyberGlove was insufficient to gather the necessary outputs to be used within the neural networks, a new calibration program had to be designed and developed. The manual calibration process

was responsible for providing the benchmark data that would be used to carry out comparisons amongst the neural networks available for each sensor. This meant that the calibration results had to be as accurate and thorough as possible to ensure a valid comparison was carried out.

The new calibration system comprised of a simple user interface that allows the user to launch the calibration application. By performing a predefined set of gestures, the calibration data for all of the sensors is gathered and stored in a set of data files. These predefined gestures detail certain angles that the user is required to achieve, or approximate. The process begins by launching the Virtual Hand LSR Analyzer tool, as shown in Figure 21, and the CyberGlove Virtual Hand Device Configuration Utility tool is then used to gather the calibration data for all the sensors, as shown in Figure 22.

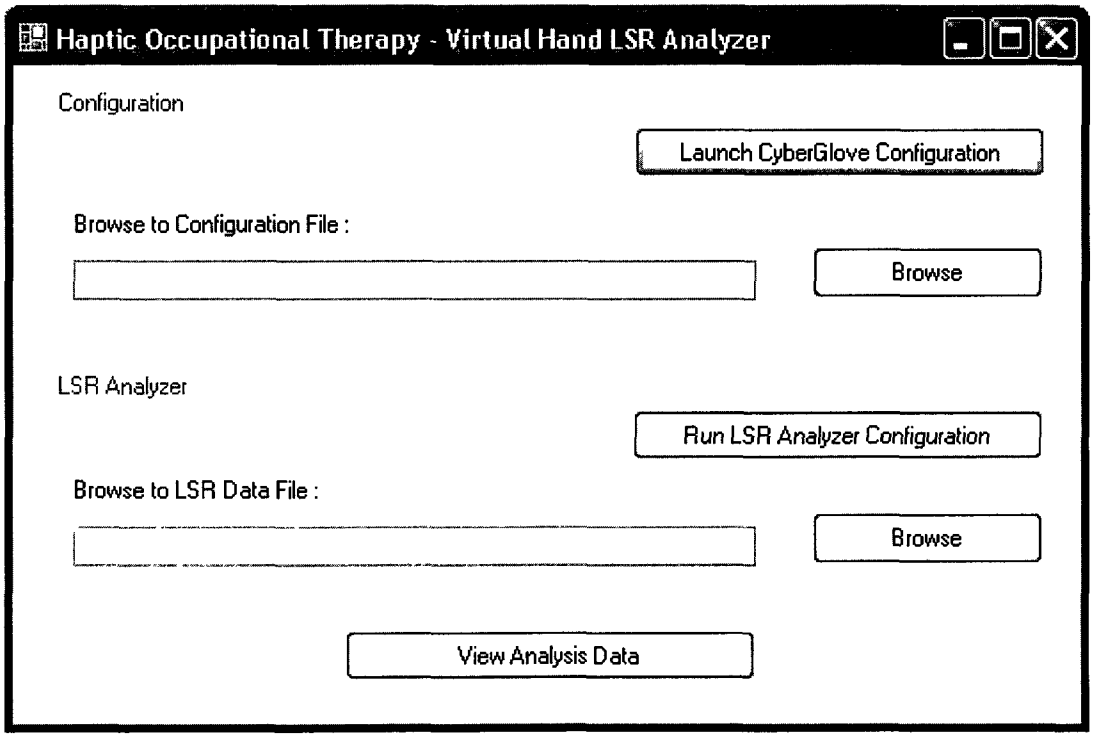


Figure 21: Virtual Hand LSR Analyzer

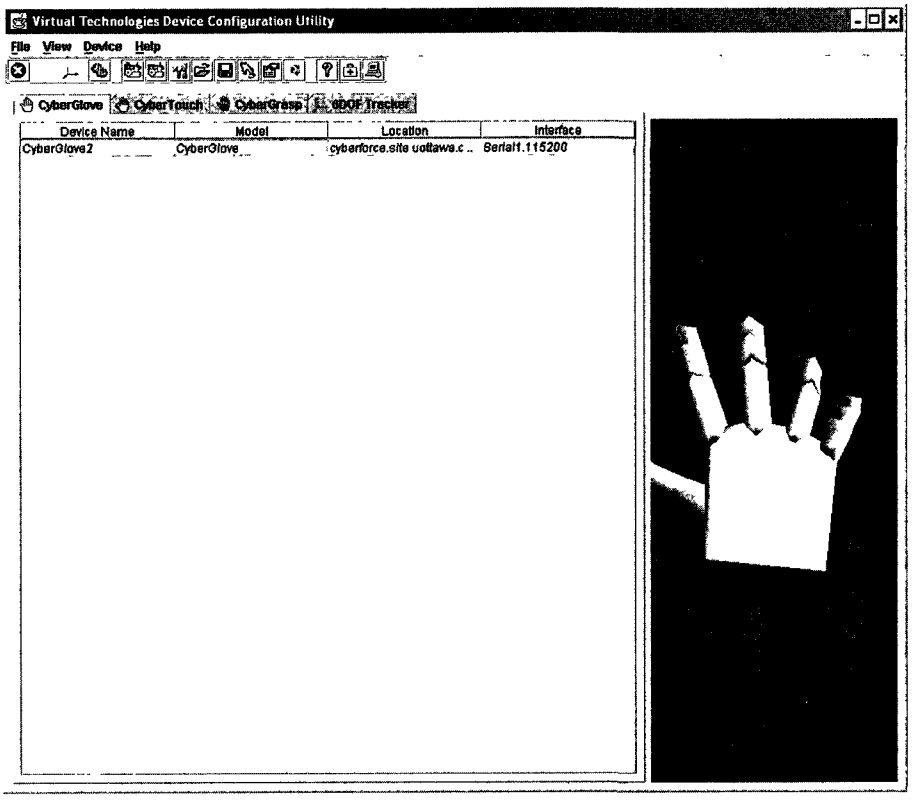


Figure 22: Virtual Hand Device Configuration Utility

After the values are gathered, including the raw sensor values, the initial Least Square Regression step is applied to produce an estimate of the gain and offset values for each sensor. For this step, equation (1) is used, where the angle $\theta$ is measured for each sensor, in each gesture performed. The angles are measured for sensors 1 to 22, shown in Figure 23.

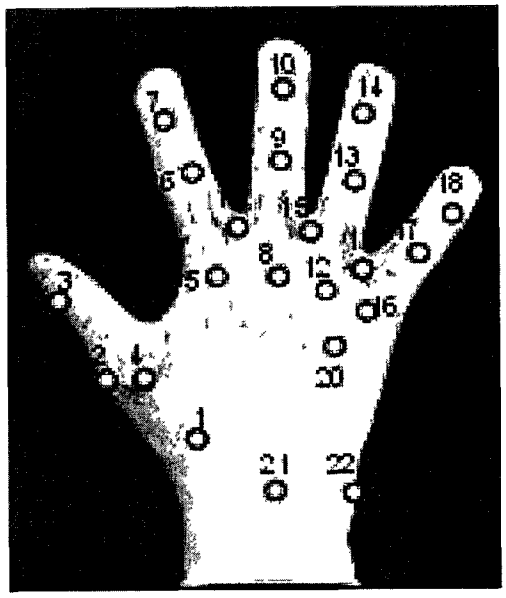


Figure 23: Instrumented CyberGlove and sensor indexing

To measure the angles, lines are drawn to represent the segments involved, and a protractor is used to measure the angle, to the nearest degree. This produces a form of uncertainty in the calculations. For this reason, more automated means of measuring angle θ will have to be determined to reduce the uncertainties in the calculations. Figure 24 shows how the angle was measured for sensor number 4, where the sensor lines were elongated to draw an exact angle that would represent the curvature of the sensor.

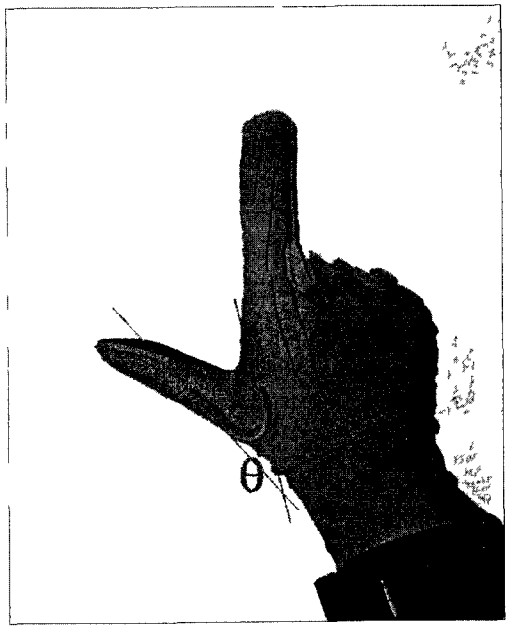


Figure 24· Measuring Angle θ for sensor number 4 for the Perpendicular Thumb gesture

Once all the angles were measured for a specific user, the values were recorded into tables that were then fed into the Least Square Regression component for analysis. A sample of the data for a user performing the perpendicular thumb gesture is shown in Table 2.

| SENSORS | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ANGLES (in degrees) | 180 | 180 | 180 | 45 | 180 | 180 | 180 | 90 | 90 | 180 | 0 |

| SENSORS | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ANGLES (in degrees) | 90 | 90 | 180 | 0 | 90 | 90 | 180 | 0 | 180 | 180 | 180 |

Table 2: Angles measured for Sensors 1 – 22 for the Perpendicular Thumb gesture for Volunteer #1

For this step in the calibration process, the data processing code used is as follows:

```
filename = 'theta1.txt';
A = textread(filename, '%d')
sensorFile = 'sensor1.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'1 %G %G %G %G %G\n', b, avr, 0, 0, 0 )
```

The data processing begins by reading the measured angles from a text file, in this case for sensor 1, followed by the sensor data from another file. The Linear Least Squares Regression method is then applied to produce the gain and the offset, and an average of all the offset values obtained is printed to the output file.

The second step of the calibration method involves determining the cross-coupling effect of the MCP flexion sensors on the abduction angles. From the first step in the calibration mechanism, we had devised a value for the gain and offset of the abduction sensors. These values, $g_{abd}$ and $r_{abd}$, can be re-applied into equation (1) to produce a calculated θ for the abduction angle, $\theta_{abd}$, such that:

$$\theta_{abd} = g_{abd}(r_{abd} - \chi) \qquad (3)$$

Once the abduction angle is calculated, a difference between the measured angle, θ, and the calculated angle $\theta_{abd}$ can be determined as:

$$\Delta\theta_{abd} = \theta_{abd} - \theta \qquad (4)$$

where the $\Delta\theta_{abd}$ is the error in the θ angle due to the cross-coupling effect of the adjoining MCP flexion sensors. In order to eliminate this effect, the cross parameters $k_l$, $k_r$ and $b$ need to be calculated using equation (2). Evaluating equation (2) and (4) will produce a final equation:

$$\Delta\theta_{abd} = k_l \chi_l + k_r \chi_r + b \qquad (5)$$

The user must now perform a new set of gestures to determine the effect of the adjacent MCP flexion sensors on the abduction angles. For each abduction angle, the user must:

1. Keep the abduction angle stable and move the right adjacent finger
2. Keep the abduction angle stable and move the left adjacent finger
3. Move both adjacent fingers of the abduction

For each gesture, the raw sensor value ($x_{abd}$ , $x_l$ , $x_r$) will be stored. The abduction sensor's angle $\theta_{abd}$ will also be measured and the set of values will be plugged into equation (2), to produce the $\Delta\theta_{abd}$. When measuring the angle of the abduction sensor, $\theta_{abd}$, between two fingers which are not in the same plane, the projection of one of the fingers on the plane perpendicular to that finger and containing the other adjacent finger is used. The angle would be the angle between the projection and the finger in the same plane. The measurement of this angle can be seen in Figure 25(a – b).

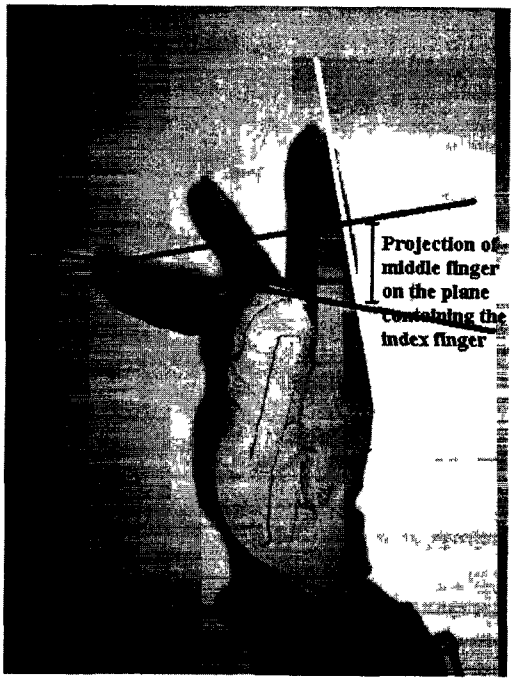


Figure 25 (a): Determining the projection of the Middle Finger on the plane containing the Index Finger.

(b) Measuring the angle between the projection of the Middle Finger and the Index Finger.

After this step, the Least Square Regression method will be used to determine the values of $k_l$ , $k_r$ and $b$. Figures 26 – 28 show the gestures required to obtain the cross parameters for the middle-ring abduction sensor, also referred to as sensor 15 in Figure 23.

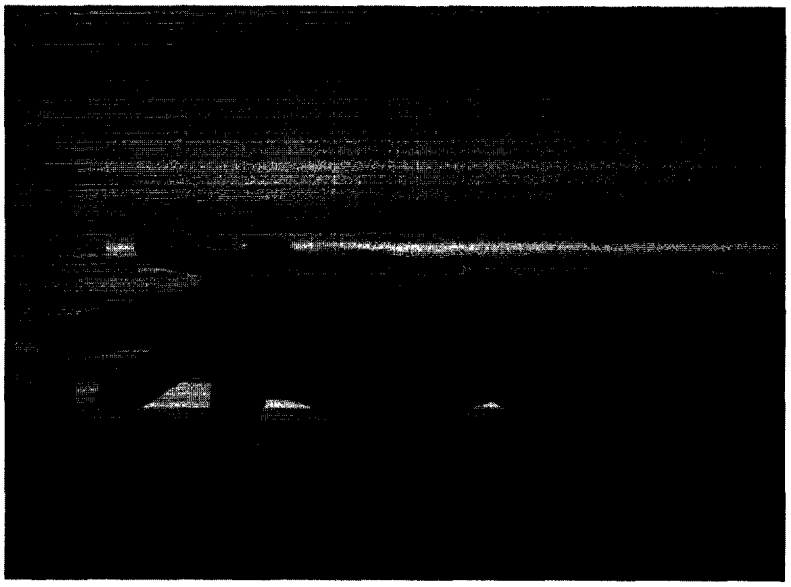

Figure 26: Moving the left adjacent finger (the middle finger)

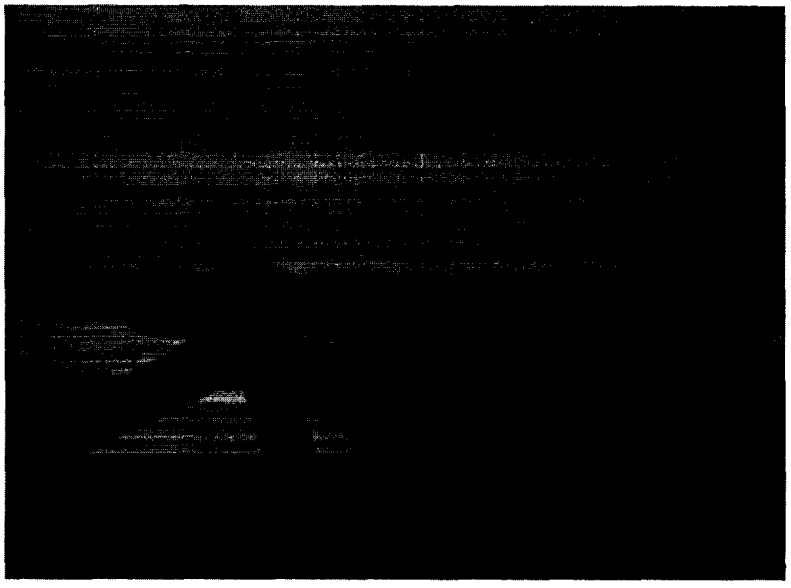

Figure 27: Moving the right adjacent finger (the ring finger)

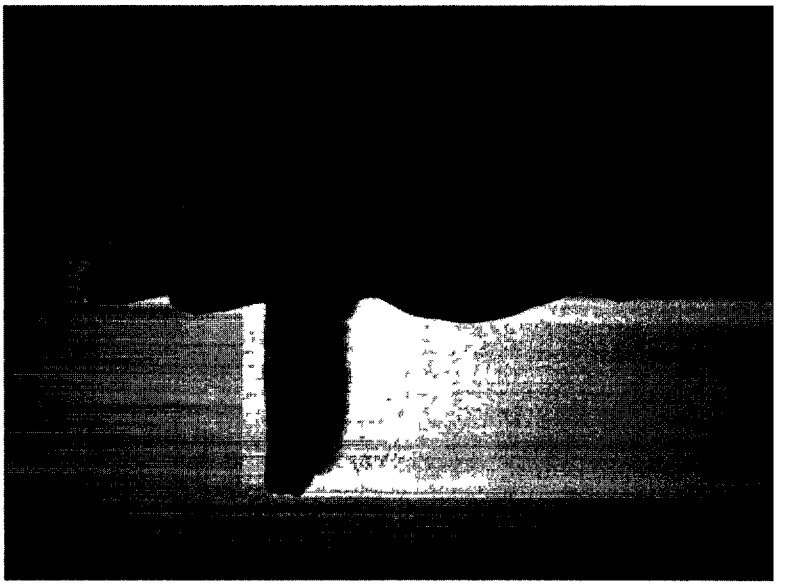

Figure 28: Moving both adjacent fingers

In order to incorporate equations (3), (4) and (5) into the data processing code, an adjustment had to be made for the processing of the abduction sensors. The amendment is as follows:

```
%sensor15
%Step 1: Calculating the Gain and Offset
filename = 'theta15.txt';
theta = textread(filename, '%d')
sensorFile = 'sensor15.txt';
sensorValues = textread(sensorFile, '%d')
[g, bint, r, rint] = regress(theta, sensorValues, 0.05)
offset = mean(r)

%Step 2: Calculating the Cross Parameters
%Part 1.a - Calculating delta theta R
realThetaFileR = 'realTheta15R.txt';
realThetaR = textread(realThetaFileR, '%d')
sensorFileR = 'sensor15R.txt';
sensorValuesR = textread(sensorFileR, '%d')
calculatedThetaR = g * sensorValuesR + offset

deltaThetaR = calculatedThetaR - realThetaR

%Part 2 - Calculating Kr and b1
%The equation is deltaTheta = kl xl + kr xr + b
%for kl = 0, deltaTheta = kr xr + b
rightSensorFile = 'sensorR15.txt';
rightSensorValues = textread(rightSensorFile, '%d')
[kr, krint, b1, b1int] = regress (deltaThetaR,
rightSensorValues, 0.05)

%Part 1.b - Calculating delta theta L
realThetaFileL = 'realTheta15L.txt';
```

```
realThetaL = textread(realThetaFileL, '%d')
sensorFileL = 'sensor15L.txt';
sensorValuesL = textread(sensorFileL, '%d')
calculatedThetaL = g * sensorValuesL + offset

deltaThetaL = calculatedThetaL - realThetaL

%Part 3 - Calculating K1 and b2
%for kr = 0, deltaTheta = k1 x1 + b
leftSensorFile = 'sensorL15.txt';
leftSensorValues = textread(leftSensorFile, '%d')
[k1, klint, b2, b2int] = regress (deltaThetaL,
leftSensorValues, 0.05)

%Part 4 - Calculating b
%b2 and b1 should be close in value.
bVal = [b1, b2]
b = mean (bVal)
b = mean (b)

%Step 3: Output all results
fprintf(outputFile,'1 %G %G %G %G %G\n', g, offset, k1, kr,
b )
```

The data processing for the second step of the calibration process requires four additional data files that will be obtained during the performance of the above-mentioned gestures. Each file will be read accordingly, with the first representing the measured abduction sensor's angles, the second being the raw data values obtained from the abduction sensor, the third is the raw data values obtained from the right adjacent flexion sensor, and the fourth is the raw data values obtained from the left adjacent flexion sensor. Then the required equations are applied and the output values of the cross parameters are printed.

Finally, a table for all the required results will be built that displays the results to the user. The calibration file produced from the LSR Analyzer component can be also used by the ANN.

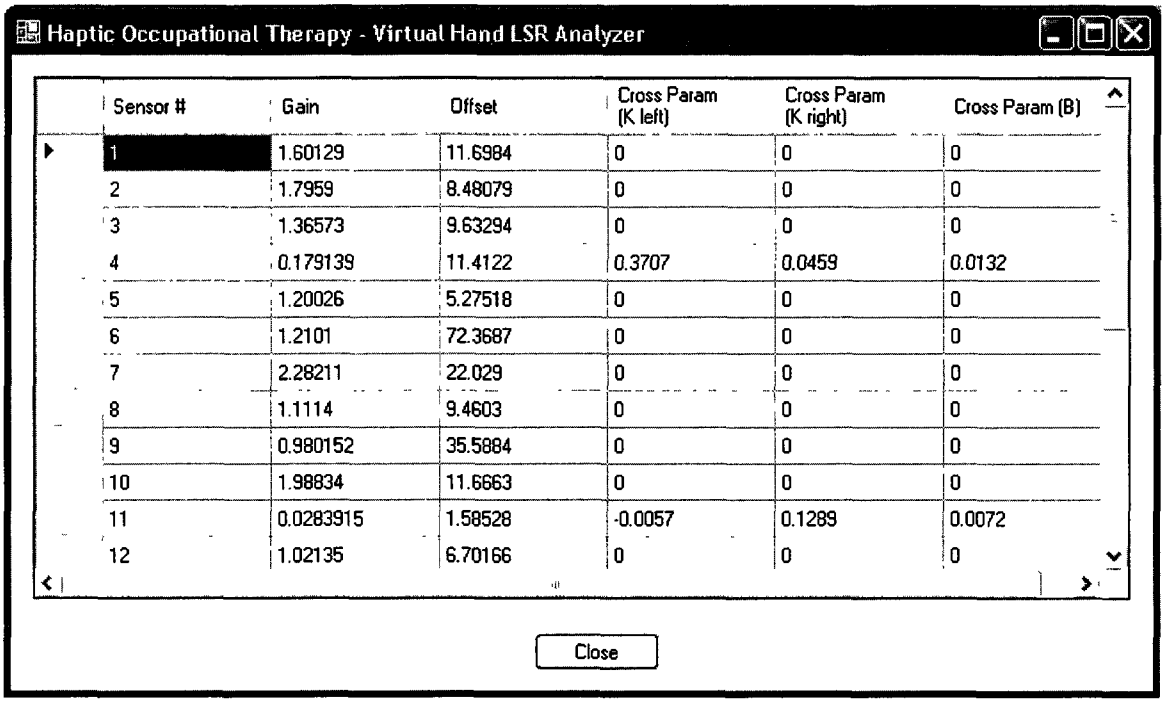


Figure 29: Screenshot of Virtual Hand LSR Analyser page displaying the final analysis results

The following table, Table 3, shows the results of the Least Square Regression method's application for a test subject:

**LSR Analysis Data - Volunteer #1**

| Sensor # | Gain | Offset | Cross Parameter-Left | Cross Parameter-Right | Cross Parameter-B |
|---|---|---|---|---|---|
| 1 | 1.60129 | 11.6984 | 0 | 0 | 0 |
| 2 | 1.7959 | 8.48079 | 0 | 0 | 0 |
| 3 | 1.36573 | 9.63294 | 0 | 0 | 0 |
| 4 | 0.179139 | 11.4122 | 0.370702 | 0.0459669 | 0.0132987 |
| 5 | 1.20026 | 5.27518 | 0 | 0 | 0 |
| 6 | 1.2101 | 72.3687 | 0 | 0 | 0 |
| 7 | 2.28211 | 22.029 | 0 | 0 | 0 |
| 8 | 1.1114 | 9.4603 | 0 | 0 | 0 |
| 9 | 0.980152 | 35.5884 | 0 | 0 | 0 |
| 10 | 1.98834 | 11.6663 | 0 | 0 | 0 |
| 11 | 0.0283915 | 1.58528 | -0.0057482 | 0.128953 | 0.00723572 |
| 12 | 1.02135 | 6.70166 | 0 | 0 | 0 |
| 13 | 0.718156 | 37.8148 | 0 | 0 | 0 |
| 14 | 1.54258 | 27.183 | 0 | 0 | 0 |
| 15 | 0.0124794 | 0.117963 | 0.0103298 | 0.0449206 | 0.00163526 |
| 16 | 0.944087 | 8.32356 | 0 | 0 | 0 |
| 17 | 0.720786 | 48.0104 | 0 | 0 | 0 |
| 18 | 1.48711 | 9.44343 | 0 | 0 | 0 |
| 19 | 0.00884842 | 0.89312 | 0.0136137 | -0.0686356 | -0.000024522 |

| | | | | | |
|---|---|---|---|---|---|
| 20 | 1.10115 | 1.69723 | 0 | 0 | 0 |
| 21 | 3.09894 | 3.19446 | 0 | 0 | 0 |
| 22 | 1.23401 | 1.29337 | 0 | 0 | 0 |

Table 3: Final results of the LSR Analyzer for Volunteer #1

## 5.2. Analysis of the Results

In order to analyze the effectiveness of the LSR Analyzer, a more comprehensive study will have to be carried out to include several volunteers, and the fully implemented Artificial Neural Network. The study will have to prove the effectiveness of using the LSR Analyzer, coupled with the ANN, in calibrating the CyberGlove and in improving its rendering within the virtual environment. Such a study is still not possible, due to third party research projects related to the implementation of the ANN.

However, by analyzing the current work carried out for the LSR Analyzer, the effectiveness of the component in revealing the discrepancies is quite visible. The initial set of data used to run the LSR Analyzer for the first step, calculating the gain and offset, contained a set of measured values for the angles of each sensor. These angles, along with the raw sensor data readings, were used to calculate the gain and offset for each sensor. However, when the same gain and offset were plugged into the LSR Analyzer again, as a part of the calculations needed for the second step of the LSR Analyzer, a $\Delta\theta_{abd}$ was determined. This was the case with each of the abduction sensors. Therefore, this comes to prove that there is indeed a difference between the actual angle measured and the angle as calculated by the CyberGlove's firmware. This leads to the conclusion that there is a need to enhance the calibration data as it does not yield exact results for the sensors.

Furthermore, upon analyzing the structure of the CyberGlove, and the organization of the sensors, it becomes evident to see that the abduction sensors undergo a twisting effect when the user moves his or her fingers. This twisting effect causes the abduction sensors to read off raw data values, even though they are not being utilized. This distortion of the abduction sensors can be tested and proven by simply laying the user's hand flat on a surface, and with the other hand, perform a twisting gesture to any of the abduction sensors on the CyberGlove. This action will cause the rendered image of the

CyberGlove in the Virtual Hand Device Configuration Utility to show a change in the position of the finger adjacent to that abduction angle. Even though there is no actual motion of the hand or the CyberGlove itself, the CyberGlove's firmware detects a change in the readings of that abduction sensor and hence renders a change in its position. Again, the mere fact that there was a $\Delta\theta_{abd}$ determined during the calculation of the cross parameters of the abduction sensors, indicates that there is in fact a difference between the actual readings and the calculated readings. Hence, the LSR Analyzer is needed to determine the difference in these two values, in order to account for a more realistic rendering of the CyberGlove and the virtual hand.

# Chapter 6

# Conclusion

In conclusion to the work presented in this thesis, there is an emphasis on the need for better quality hardware that allows for more precision in the sensor readings. There is also an essential need for hardware that adapts to the various shapes and sizes of the human hands. The current available CyberGlove can only fit the larger human hand size, while users with smaller hands will not be able to properly use the device. This causes problems when rendering the virtual hand image, as sensor readings will not reflect the actual movement of the user's hand. The other problem that exists is with the CyberForce system, where the range of motion is limited by the size of the hardware itself. The CyberArm has a somewhat short wire that prevents the user from moving his or her arm freely. The limiting factors also include the joints in the CyberArm.

The limitations of the hardware are beyond the physical aspects where there was a need to modify the calibration process. This was necessary as the calibration process imposed impractical requirements on the user which some stroke patients were deemed incapable of performing. The other reason was due to the method of calibration itself. The virtual rendering of the user's hand reflected improper actions and gestures. This was due to the evaluation of the raw sensor readings of the abduction sensors, where the readings of the adjacent flexion sensors were not taken into consideration. This meant that the direction of the movements of the abduction angles was not accurately reflected. Thus, the calibration process needed to be enhanced with the use of a Least Squares Regression method, and the application of an Artificial Neural Network to the system. The ANN meant that patients did not have to go through a tedious and difficult calibration process that would have been daunting for patients with upper extremity problems. It also meant that periodic calibrations were possible and quick to complete, as patients' performance improved.

The implementation of a progress measurement component was a novel idea in the field of virtual rehabilitation. It allowed for patients to monitor their own progress and to advance in training levels without the constant monitoring of an Occupational Therapist. This ability would reduce the amount of time needed by therapists in analyzing the performance of their patients and provide them with more time to concentrate on the more needy patients. It also allows the patients to set their own goals and the ability to achieve those goals at their own pace.

During the research performed for this thesis, the use of actual Occupational Therapists was deemed highly useful and beneficial. The feedback provided by the therapists allowed for the creation of more realistic and useful exercises that were more meaningful to a patient in his or her everyday life. These exercises meant that patients could learn to become more self-sufficient at a faster rate, as they were improving their abilities to perform actions that were necessary from day to day. These included actions of eating, drinking and reaching for common kitchen objects. The Occupational Therapists feedback also delved into the practicality of the system. Since the therapists did not come from a computer technical background, they found the presence of user interfaces very helpful in guiding them through the use of the system, without having to go through long and tedious user manuals. It would thus be fair to conclude that more cooperation and interaction with specialists in the field of Occupational Therapy would allow developers and researchers to design better virtual rehabilitation systems that would tailor to the need of the patients.

## 6.1. Future Work

As with any research project, the work here is not yet complete. There is much room for improvements and many new requirements that can enhance the system into a more complete functional unit. One of the more significant aspects worth improving is the hardware that is currently in use. There are a limited number of products in the market that allow for full arm haptic control. Many of the well developed products available deal with either several digits of the hand, or with the arm separately. These devices do not cover the requirements of the Virtual Rehabilitation system, where there

is a need to reflect the full range of motion of the human arm in space to capture the movements of the human hand.

Other enhancements of the system include designing a dynamic virtual hand image that is built based on the user's actual hand. The size of the hand can be determined by the calibration process for the Artificial Neural Network, when the user's hand is placed on the chess board. Another enhancement would be to synchronize the setup of the system in such a way that it becomes easy, fast and efficient to install the system in rehabilitation centers. The process currently involves several steps for gathering data before the initial setup of the system. These steps must be improved to become an integrated part of the system, such that the system's deployment will not require any data gathering.

The enhanced calibration process of the virtual hand has much room for improvements as well. The angles for each sensor are currently being measured manually. This method introduces for a significant error range for the measured angles' values. To reduce these errors, a method for automatically measuring the angles, to the nearest tenth of a degree, will be necessary.

The other improvement needed for the calibration process is with the use of the data files. The LSR analyzer requires over thirty data files, which are currently being written by hand. The Virtual Hand Device Calibration Unit currently does not produce an output file with the required data for each gesture performed. To eliminate this tedious and monotonous task, the virtual hand application must be enhanced to produce the required data values into an output file, as the user performs the required gestures. These output files will then have to be formatted by the Virtual Hand LSR Analyzer tool to produce temporary input files that will be fed into the LSR analysis code. The most significant enhancement needed here is the modification of the Virtual Hand Device Calibration Unit to function as required by the Virtual Hand LSR Analyzer application.

The field of haptics is currently overrun by researches into tele-haptic applications. However, the main concern is with the protocol used to transmit the data, and with the processing of this data at the destination nodes. Once tele-haptics is a resolved problem, tele-rehabilitation will become the new method of rehabilitating patients. It will allow patients and occupational therapists to cooperate across continents, allowing patients for more time with an occupational therapist than is currently permitted by the limitations of locality.

The final enhancement to be made to the system would be an improvement of the new calibration process. The work of B. Wang et al. [18], elaborated on the method of calibrating the thumb-index abduction sensor, where the thumb roll sensor was used to adjust the former sensor's calibration. The method used in this thesis relied only on the adjacent two flexion sensors, the thumb flexion sensor and the index flexion sensor. This meant that any effects of the thumb roll sensors were not accounted for throughout the calibration method used. To modify this work, a few more gestures would be needed to gather the data, as per B. Wang et al. [18], and three more equations would have to be used to replace the ones currently in use. This work should allow for a better determination of the cross parameters affecting the thumb-index abduction sensor.

The final addition to the system would be actual trials in a rehabilitation centre, where patients with deformities of the upper extremities can test the effectiveness of the Virtual Haptic Motor Rehabilitation System. The trials will determine if the patients will improve in their abilities to perform daily-life activities, and if they will gain more control over their upper extremities. They will also expose any areas of further improvements that could help enhance the system's effectiveness and help patients achieve a higher level of success.

# References

[1] P. Rijken and J. Dekker, "Clinical Experiences of Rehabilitation Therapists with Chronic Diseases: A Quantitative Approach," Clinical Rehabilitation, vol. 12, pp. 143-150. 1998.

[2] A. A. Rizzo and J. G. Buckwalter, "Virtual Reality And Cognitive Assessment And Rehabilitation: The State of the Art", in Virtual Reality in Neuro-Psycho-Physiology, pp. 123–146. Ed. Amsterdam, The Netherlands: IOS, 1997.

[3] G. Riva, "Virtual Reality in Psychotherapy: Review," CyberPsychology & Behavior. Vol.8, No.3, pp. 220-230, June 2005.

[4] Wikipedia (October 4, 2009), *Virtual Rehabilitation,* Retrieved October 12, 2009, from http://en.wikipedia.org/wiki/Virtual_rehabilitation

[5] B.B.C. (June 10, 2004), *'Virtual' boost for stroke patients*, Retrieved October 12, 2009, from http://news.bbc.co.uk/2/hi/uk_news/northern_ireland/3791779.stm

[6] V3.co.uk News (August 31, 2006), *Medical marvel created from Xbox*, by Robert Jacques, Retrieved October 13, 2009, from http://www.v3.co.uk/vnunet/news/2163228/medical-marvel-created-xbox

[7] Wikipedia, *Haptic Technology,* Retrieved October 14, 2009, from http://en.wikipedia.org/wiki/Haptic_technology

[8] Immersion Medical Products, Retrieved October 13, 2009, from http://www.immersion.com/markets/medical/products/index.html

[9] Immersion Products, *TouchSense® Force Feedback System – 6000 Series for Gaming*, Retrieved October 14th, 2009, from http://www.immersion.com/products/arcade-electronics/

[10] M. McLaughlin, A. Rizzo, Y. Jung, W. Peng, S. Yeh, W. Zhu, and the USC/UT Consortium for Interdisciplinary Research, "Haptics-enhanced virtual environments for stroke rehabilitation", Proceedings of the IPSI, Cambridge, MA. 2005.

[11] George V. Popescu, "Design and Performance Analysis of a Virtual Reality-based Telerehabilitation System", New Brunswick, New Jersey, January 2001.

[12] J. Zhou, F. Malric, S. Shirmohammadi, "A New Hand Measurement Method to Simplify Calibration in CyberGlove-BasedVirtual Rehabilitation", IEEE Transactions on

Instrumentation and Measurement, (accepted, to appear), 2009

[13] S., Yeh, A., Rizzo, W., Zhu, J., Stewart, M., McLaughlin, I., Cohen, Y., Jung, and W., Peng,"An integrated system: virtual reality, haptics and modern sensing technique (VHS) for post-stroke rehabilitation", proceedings of the ACM symposium on Virtual reality software and technology', New York, 2005.

[14] D., Jack, R., Boian, A.., Merians, M., Tremaine, G., Burdea, S., Adamovich, M., Recce, and H., Poizner. "Virtual reality-enhanced stroke rehabilitation" IEEE Trans. on Neurological Sys. and Rehabilitation Engineering, Vol. 9, pp. 308-318, 2001.

[15] R., Boian, A., Sharma, C., Han, A., Merians, G., Burdea, S., Adamovich, M., Recce, M., Tremaine, H., Poizner,"Virtual Reality-Based Post-Stroke Hand Rehabilitation", Proc. Medicine Meets Virtual Reality 2002 Conference, CA, January, 2002.

[16] J., Broeren, A., Bjorkdahl, R., Pascher, and M., Rydmark, "Virtual Reality and Haptics as an Assessment Device in the Postacute Phase after Stroke", CyberPsychology & Behavior, Vol. 5, No. 3, 2002, pp. 207-211.

[17] A., Alamri, R., Iglesias, M., Eid, A., El Saddik, S., Shirmohammadi, and E., Lemaire, "Haptic-based Exercises for Post-Stroke Patient Rehabilitation," IEEE Workshop on Medical Measurement and Applications, Warsaw, Poland, 4-5 May, 2007.

[18] B. Wang, and S. Dai, "Dataglove calibration with constructed grasping gesture database", *IEEE International Conference on Virtual Environment, Human-Computer Interfaces and Measurements Systems (VECIMS, 09)*, Hong Kong, China, May 11-13, 2009, pp. 6-11

[19] Wikipedia, *Least squares,* Retrieved October 21, 2009, from http://en.wikipedia.org/wiki/Least_squares

[20] P. L. Weiss, P. Bialik and R. Kizony, "Virtual Reality Provides Leisure Time Opportunities for Young Adults with Physical and Intellectual Disabilities," CyberPsychology and Behavior, Volume 6, pp. 335-342, 2003.

[21] H. Oyama, F. Wakao, "Evaluation of a virtual reality system for medicine", *Proceedings of the 1997 International Conference on Virtual Systems and Multimedia*, pp. 243, September 1997

[22] U. Meier, F.J. García, N.C. Parr, C. Monserrat, J.A. Gil, V. Grau, M.C. Juan and M. Alcañiz, "3D Surgery Trainer with Force Feedback in Minimally Invasive Surgery,"

International Congress Series, Volume 1230, pp. 32-37, June 2001.

[23] *The Rutgers Ankle* (Updated May 23, 2001), Retrieved September 12, 2009 from http://www.caip.rutgers.edu/vrlab/projects/ankle/ankle.html

[24] Burdea, G., Popescu V., V. Hentz, and K. Colbert, "Virtual Reality-based Orthopedic Tele-rehabilitation," IEEE Transactions on Rehabilitation Engineering, Vol. 8, No. 3, pp. 429-432, September 2000

[25] R. F., Boian, J. E., Deutsch, C. S., Lee, G. C., Burdea, and J., Lewis, "Haptic Effects for Virtual Reality-Based Post-Stroke Rehabilitation," Proc. of 11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003.

[26] J., Deutsch, J., Latonio, G., Burdea, and R., Boian,"Rehabilitation of Musculoskeletal Injuries Using the Rutgers Ankle Haptic Interface: Three Case Reports", in proceedings of EuroHaptics 2001 Conference. UK, July, 2001.

[27] M. Kuttuva, R. Boian, A. Merians, G. Burdea, M. Bouzit, J. Lewis, and D. Fensterheim, "The Rutgers Arm: An Upper-Extremity Rehabilitation System in Virtual Reality", Int.Workshop on Virtual Rehabilitation (IWVR05), Catalina Island, California, pp. 94-103, September 2005

[28] G. Burdea, D. Fensterheim, D. Cioi and A. Arezki, "The Rutgers Arm II Rehabilitation System", Proceedings of Virtual Rehabilitation 2008, pp. 76, Vancouver Canada, August 25-27, 2008

[29] R., Colombo, F., Pisano, S., Micera, A., Mazzone, C., Delconte, M., Carrozza, P., Dario, and G., Minuco,"Upper limb rehabilitation and evaluation of stroke patients using robot-aided techniques," proceeding of 9th International Conference on Rehabilitation Robotics, Chicago, June, 2005.

[30] R., Loureiro, F., Amirabdollahian, S., Coote, E., Stokes, W., Harwin,"Using Haptics Technology to Deliver Motivational Therapies in Stroke Patients: Concepts and Initial Pilot Studies", Proc. EuroHaptics 2001 Conference. UK, July, 2001.

[31] J., Furusho, C., Li, Y., Yamaguchi, S., Kimura, K. Nakayama, T., Katuragi, T., Oguri, U., Ryu, S., Suzuki, and A., Inoue,"A 6-DOF rehabilitation machine for upper limbs including wrists using ER actuators", in proceedings of IEEE International Conference of Mechatronics and Automation, 2005.

[32] H. Sveistrup, J. McComas, M. Thornton, S. Marshall, H. Finestone, A. McCormick,

K. Babulic and A. Mayhew, "Experimental Studies of Virtual Reality-Delivered Compared to Conventional Exercise Programs for Rehabilitation," CyberPsychology and Behavior, Volume 6, pp. 245-249, 2003.

[33] H. Sveistrup, "Motor rehabilitation using virtual reality," Journal of Neuroengineering and Rehabilitation, Dec 10, 2004.

[34] Broeren J, Björkdahl A, Pascher R, Rydmark M, "Virtual reality and haptics as an assessment device in the postacute phase after stroke", *Cyberpsychol Behav* 2002, **5:**207-211

[35] Bardorfer A, Munih M, Zupan A, Primozic A, "Upper limb motion analysis using haptic interface", *IEEE/ASME Transactions on Mechatronics*, 2001, **6:**253-260

[36] I. Shakra, M. Orozco, A. El Saddik, S. Shirmohammadi, and E. Lemaire, "Haptic Instrumentation for Physical Rehabilitation of Stroke Patients", In proceedings of the 2006 IEEE International Workshop on Medical Measurement and Applications Benevento, Italy, April 2006.

[37] R.H. Jebsen, N. Taylor, R.B. Trieschmann, M.J. Trotter, and L.A. Howard, "An Objective and Standardized Test of Hand Function," Archives of Physical Medicine and Rehabilitation, pages 311-319. June 1969.

[38] V. Mathiowitz, G. Volland, N. Kashman, and K. Weber, "Adult Norms for the Box and Blocks Test of Manual Dexterity," The American Journal of Occupational Therapy, pages 386-391. 1985.

[39] R. Kayyali, S. Shirmohammadi, A. El Saddik, E. Lemaire, "Daily-Life Exercises for Haptic Motor Rehabilitation", IEEE International Workshop on Haptic Audio Visual Environments and their Applications, Ottawa, Canada, October 2007

[40] R. Kayyali, S. Shirmohammadi, "A benchamrked automated progress measurement system for haptic motor rehabilitation", International Journal of Advanced Media and Communication, Vol. 3, Number 1-2 / 2009, pp. 179 - 196

[41] W. B. Griffin, R. P. Findley, M. L. Turner, and M. R. Cutkosky, "Calibration and mapping of a human hand for dexterous telemanipulation," in Proc. of the ASME IMECE Dynamics Systems and Controls Division, vol. 69, 2000, pp. 1145-1152

[42] A. S. Menon, B. Barnes, R. Mills, C. D. Bruyns, E. Twombly, J. Smith, K. Montgomery, and R. Boyle, "Using registration, calibration, and robotics to build a more

accurate virtual reality simulation for astronaut training and telemedicine", in *Proc. Of WSCG 2003,* Czech Republic, Feb. 3-7, 2003

[43] M. Fischer, P. Smagt, and G. Hirzinger, "Learning techniques in a dataglove based telemanipulation system for DLR hand", in Proc. Of the 1998 IEEE International Conference on Robotics & Automation, Leuven, Belgium, May 1998, pp. 1603-1608

[44] F. Kahlesz, G. Zachmann, and R. Klein, "Visual-fidelity dataglove calibration", in Proc. of the Computer Graphics International *(CGI'04)*, 2004, pp. 403-410

[45] K. Kahol, P. Tripathi, and S. Panchanathan, "Recognizing everyday human movements through human anatomy based coupled hidden markov model", *International* Journal on Systemics, Cybernetics and Informatics, Pentagram Publications, India, 2005

[46] D. Choi, S. Oh, H. Chang, and K. Kim, "Non-linear camera calibration using neural networks", Neural, Parallel & Scientific Computations, vol. 2, no. 1, pp. 29-42, Mar. 1994

[47] R. Anchini, C. Liguori, V. Paciello, and A. Paolillo, "A comparison between stereo-vision techniques for the reconstruction of 3-D coordinates of objects", *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1459-1466, Oct. 2006

[48] Y. Liu, "Calibrating an industrial microwave six-port instrument using the artificial neural network technique", *IEEE Transactions on Instrumentation and Measurement*, vol. 45, no. 2, pp. 651-656, Apr. 1996.M. L

[49] J. M. Dias Pereira, O. Postolache, P. M. B. Silva Girão, and M. Cretu, "Minimizing temperature drift errors of conditioning circuits using artificial neural networks", *IEEE Transactions on Instrumentation and Measurement*, vol. 49, no. 5, pp. 1122-1127, Oct. 2000

[50] R. Kayyali, A. Elamri, M. Eid, R. Iglesias, S. Shirmohammadi, A. El Saddik, E. Lemaire, "Occupational Therapists' Evaluation of Haptic Motor Rehabilitation", *Proc. EMBC Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Lyon, France 2007

[51] Wikipedia, *Artificial neural network*, Retrieved December 9, 2009, from http://en.wikipedia.org/wiki/Artificial_neural_network

[52] Wikipedia, *Linear regression*, Retrieved December 13, 2009, from

http://en.wikipedia.org/wiki/Linear_regression

[53] Wikipedia, *Degrees of freedom (mechanics)*, Retrieved December 13, 2009, from http://en.wikipedia.org/wiki/Degrees_of_freedom_(mechanics)

[54] G. Burdea and P. Coiffet, *Virtual reality technology*, Hoboken, NJ: John Wiley & Sons, 2003.

[55] Broeren J, Lundberg M, Molen T, Samuelsson , Sunnerhagen KS, Bellner A, Rydmark M, "Virtual reality and haptics as an assessment tool for patients with visuospatial neglect: a preliminary study", *In Proceedings of the Second International Workshop on Virtual Rehabilitation*: Piscataway. Edited by: Burdea GC, Thalmann D. Lewis JA: IWAR2003; 2003:27-32. September 21–22 2003

[56] WordNet Search 3.0, *Metacarpal,* Retrieved March 23, 2010, from http://wordnetweb.princeton.edu/perl/webwn?s=metacarpal

[57] WordNet Search 3.0, *Palmar,* Retrieved March 23, 2010, from http://wordnetweb.princeton.edu/perl/webwn?s=palmar

[58] WordNet Search 3.0, *Dorsal,* Retrieved March 23, 2010, from http://wordnetweb.princeton.edu/perl/webwn?s=dorsal

[59] Wikipedia, *Occupational Therapist*, Retrieved March 23, 2010, from http://en.wikipedia.org/wiki/Occupational_Therapist

[60] Heart and Stroke Foundation, *Statistics,* Retrieved March 16, 2010, from http://www.heartandstroke.com/site/c.ikIQLcMWJtE/b.3483991/k.34A8/Statistics.htm#stroke

[61] Wikipedia, *Stroke*, Retrieved March 16, 2010, from http://en.wikipedia.org/wiki/Stroke

# Appendix A – Hand Gestures for Virtual Hand Calibration

## Calibration - Part 1

### Open Palm

Figure 30: Dorsal View of Open Palm gesture

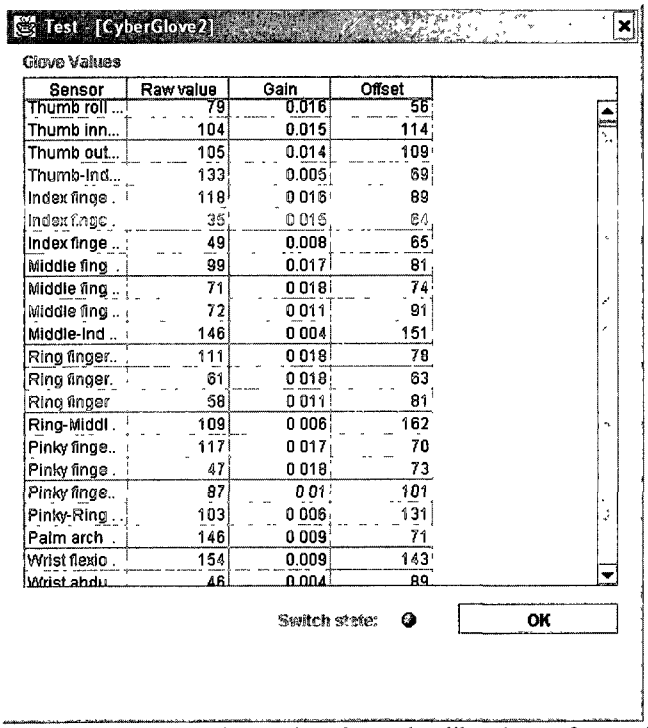


Figure 31: Data captured from the CyberGlove Virtual Hand calibration software for Open Palm gesture

## Open Hand

Figure 32: Dorsal View of Open Hand gesture

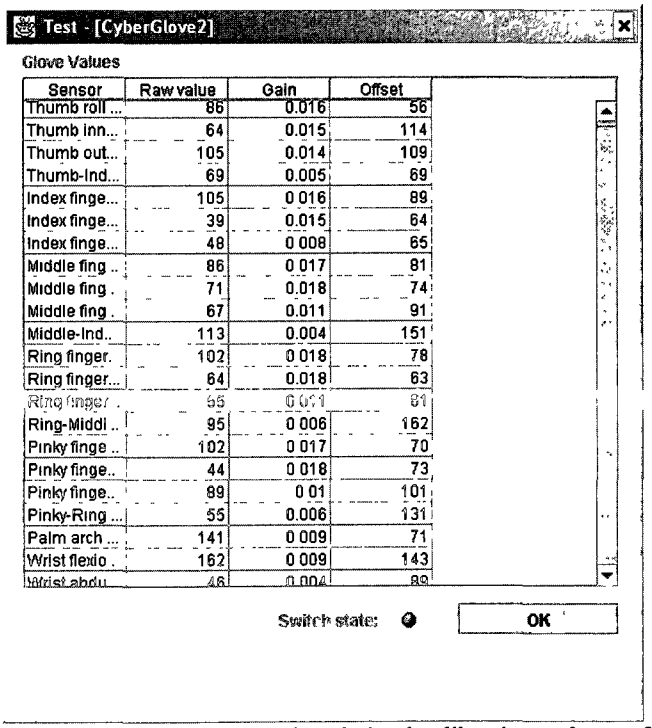


Figure 33: Data captured from the CyberGlove Virtual Hand calibration software for Open Hand gesture

## Perpendicular Thumb

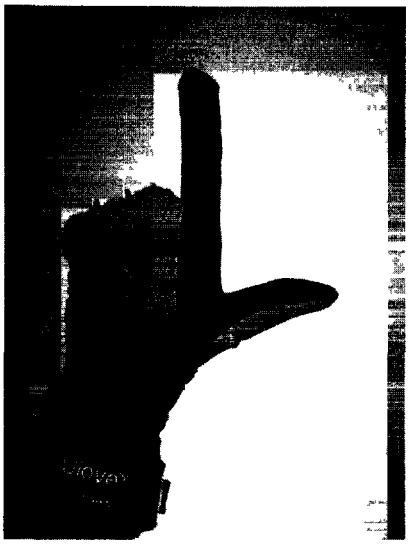

Figure 34: Palmar View of Perpendicular Thumb gesture

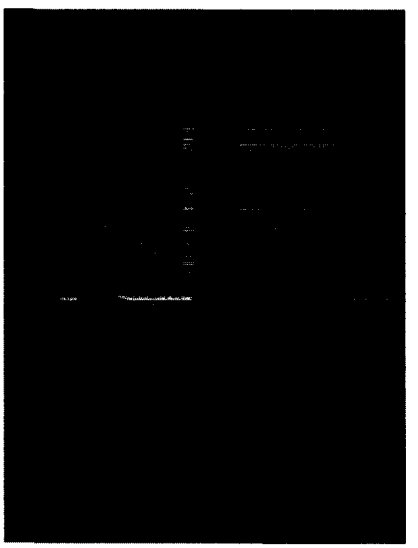

Figure 35: Dorsal View of Perpendicular Thumb gesture

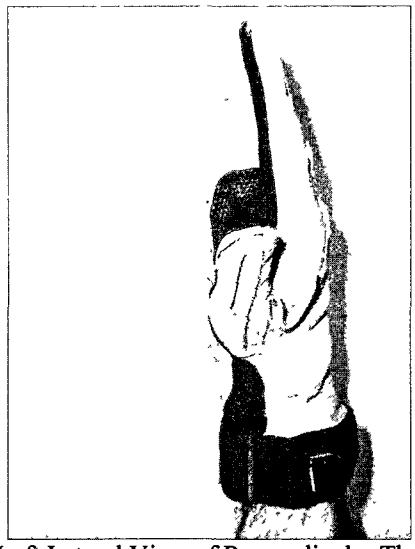

Figure 36: Left Lateral View of Perpendicular Thumb gesture

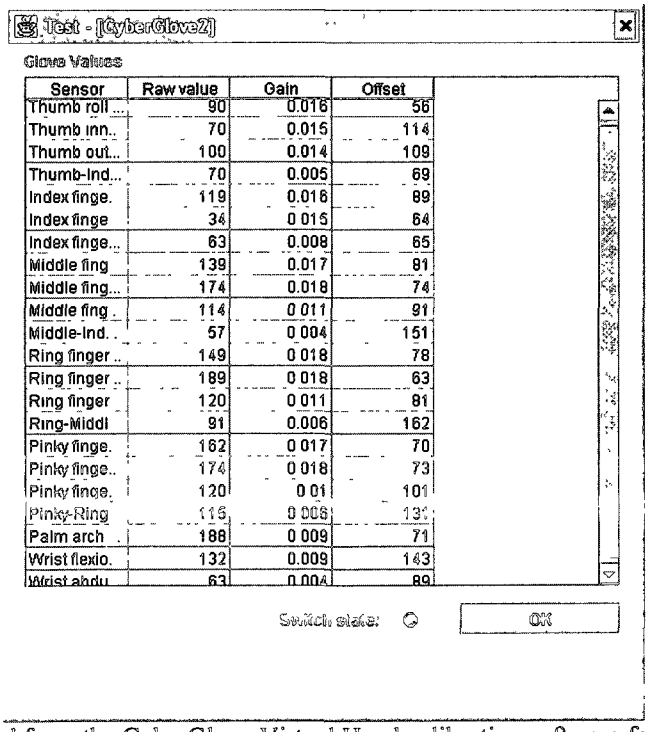


Figure 37: Data captured from the CyberGlove Virtual Hand calibration software for Perpendicular Thumb gesture

## Closed Fist

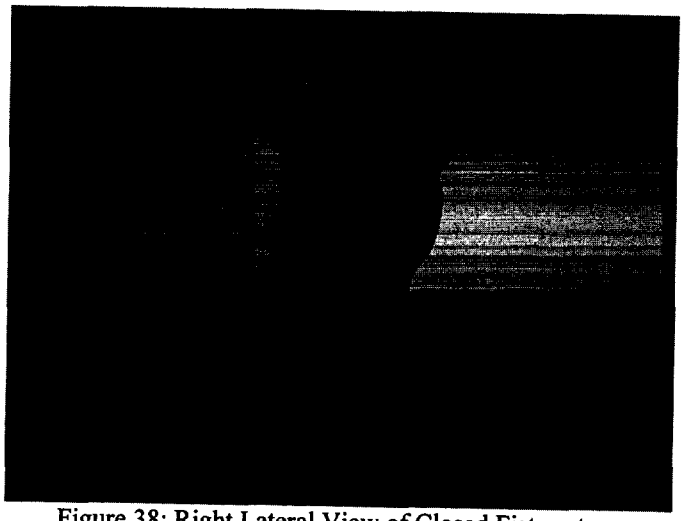

Figure 38: Right Lateral View of Closed Fist gesture

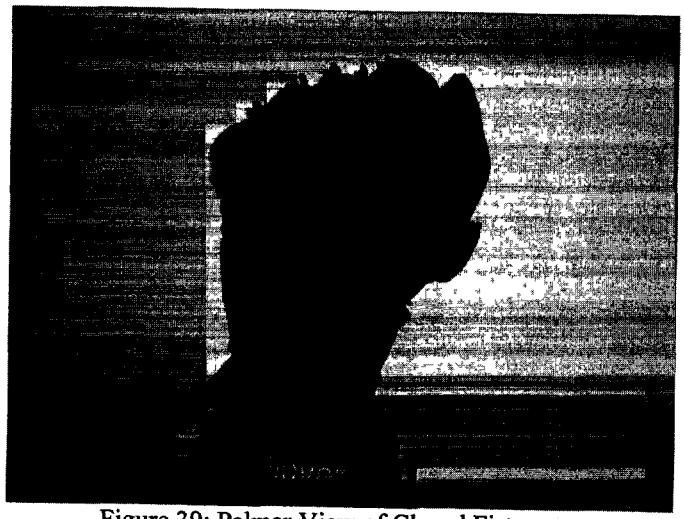

Figure 39: Palmar View of Closed Fist gesture

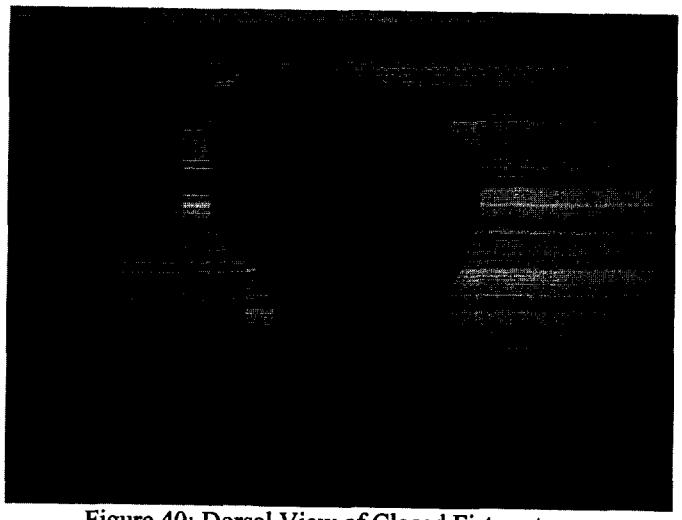

Figure 40: Dorsal View of Closed Fist gesture

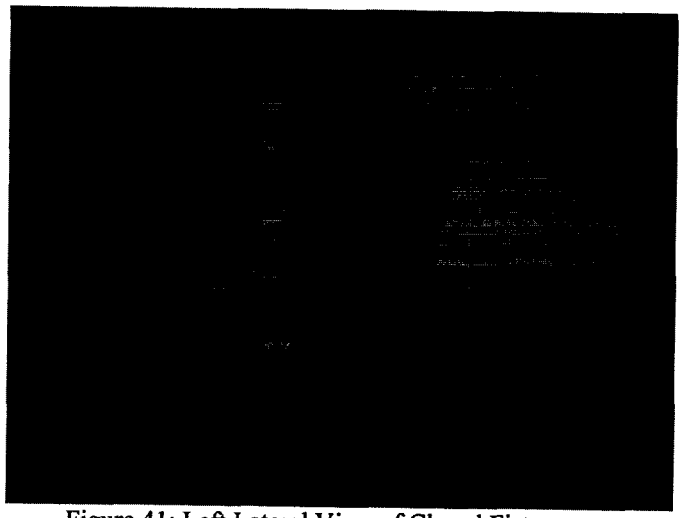

Figure 41: Left Lateral View of Closed Fist gesture

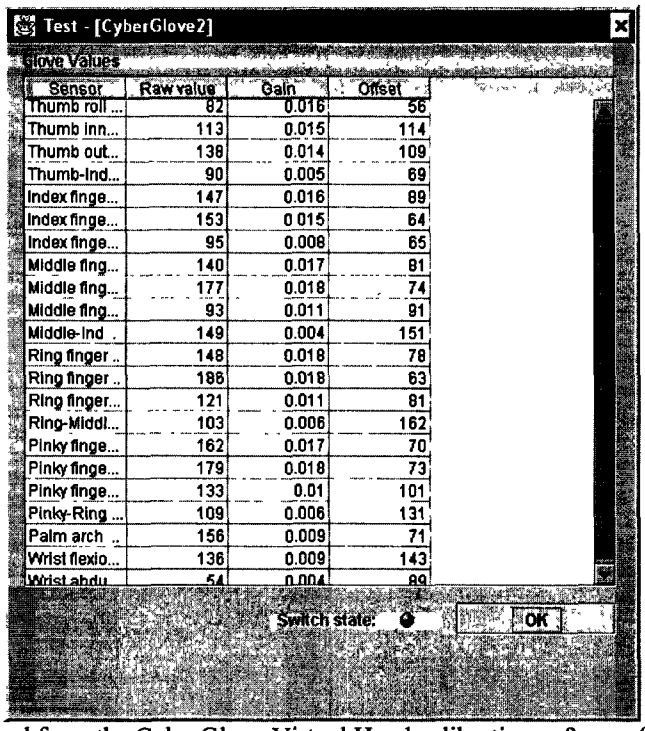

Figure 42: Data captured from the CyberGlove Virtual Hand calibration software for Closed Fist gesture

## Thumb's Up

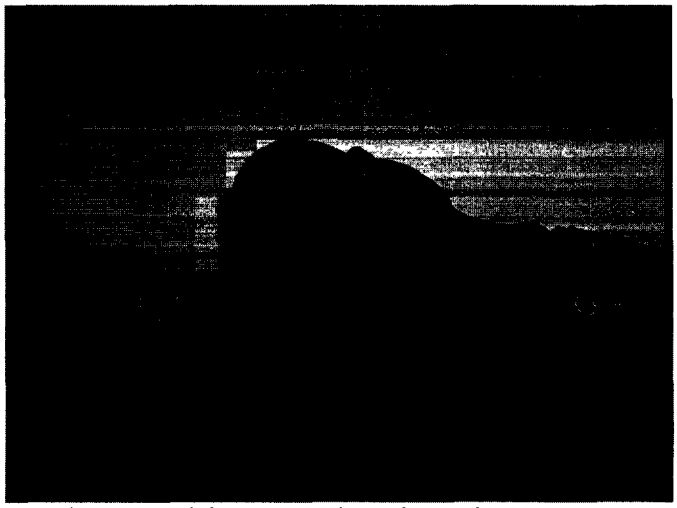
Figure 43: Right Lateral View of Thumb's Up gesture

Figure 44: Dorsal View of Thumb's Up gesture

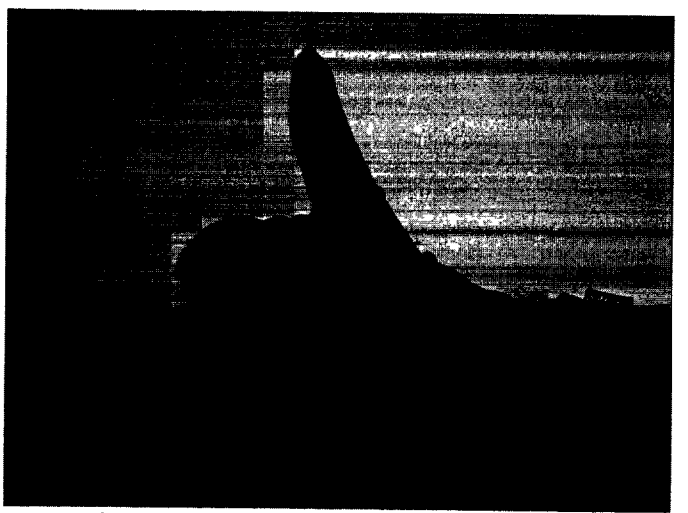

Figure 45: Palmar View of Thumb's Up gesture

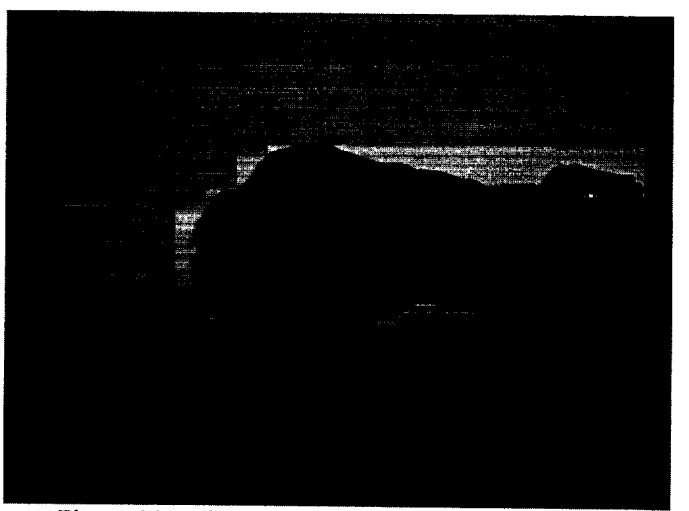

Figure 46: Left Lateral View of Thumb's Up gesture

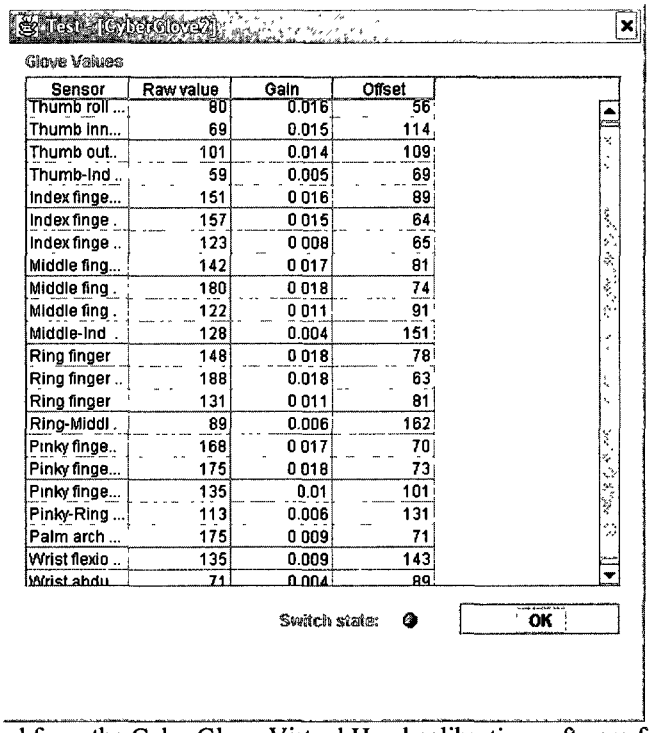


Figure 47: Data captured from the CyberGlove Virtual Hand calibration software for Thumb's Up gesture

## Perpendicular Palm

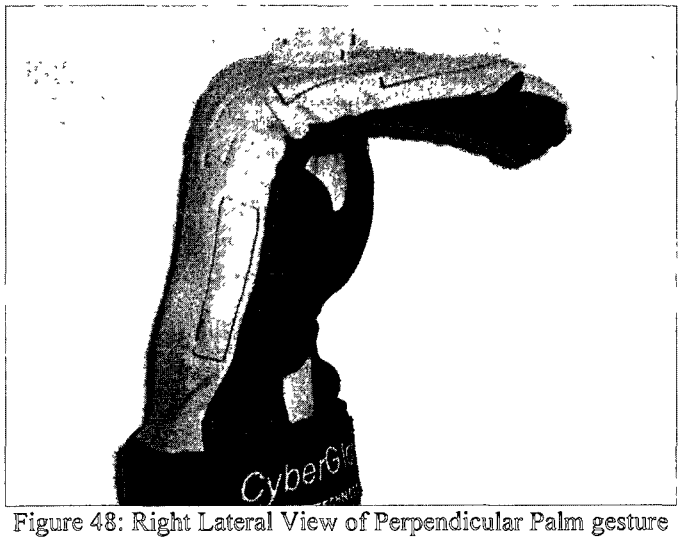

Figure 48: Right Lateral View of Perpendicular Palm gesture

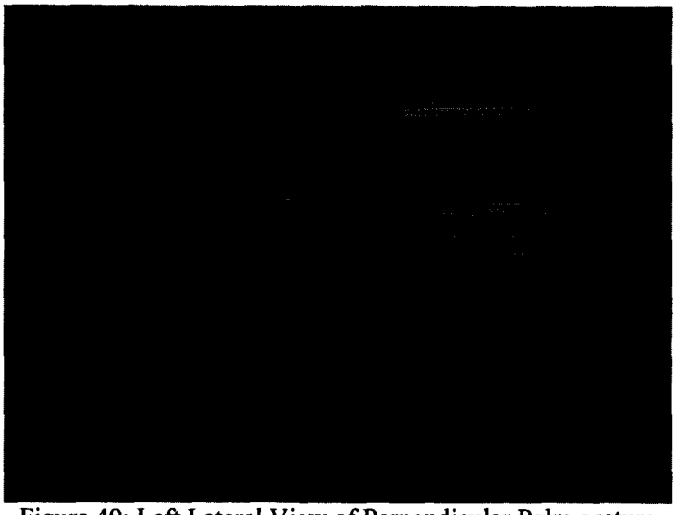

Figure 49: Left Lateral View of Perpendicular Palm gesture

Figure 50: Palmar View of Perpendicular Palm gesture

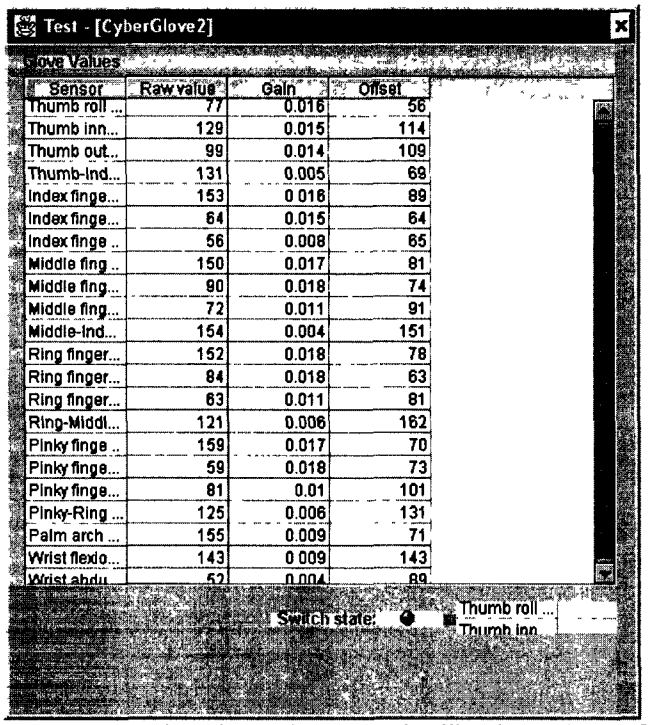

Figure 51: Data captured from the CyberGlove Virtual Hand calibration software for Perpendicular Palm gesture

## Circular Palm

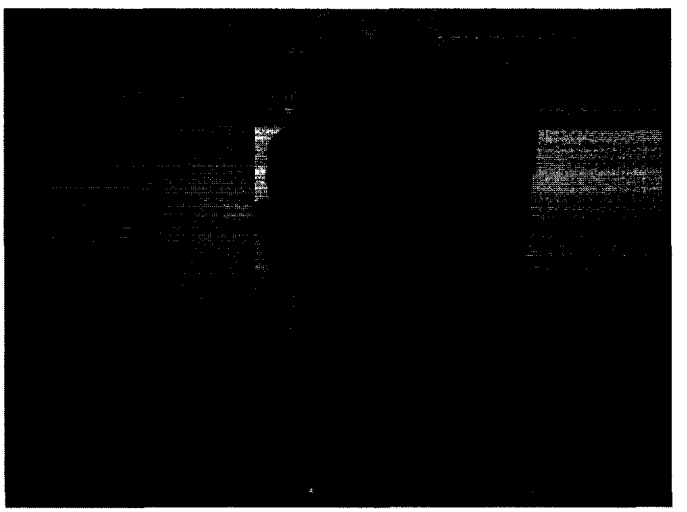
Figure 52: Dorsal View of Circular Palm gesture

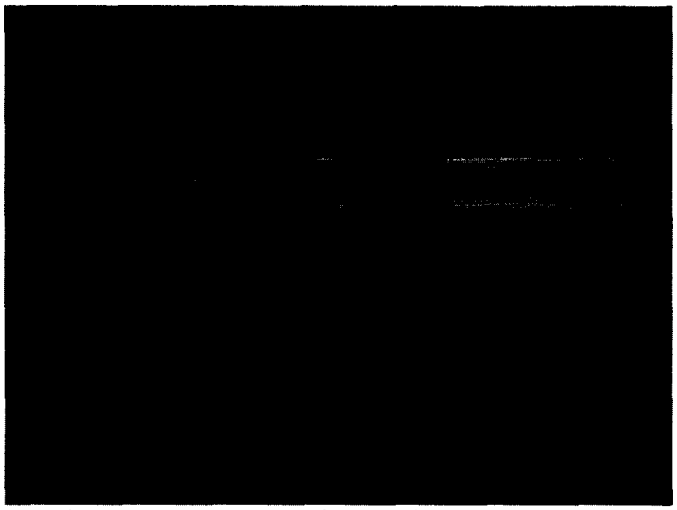

Figure 53: Left Lateral View of Circular Palm gesture

Figure 54: Palmer View of Circular Palm gesture

Figure 55: Right Lateral View of Circular Palm gesture

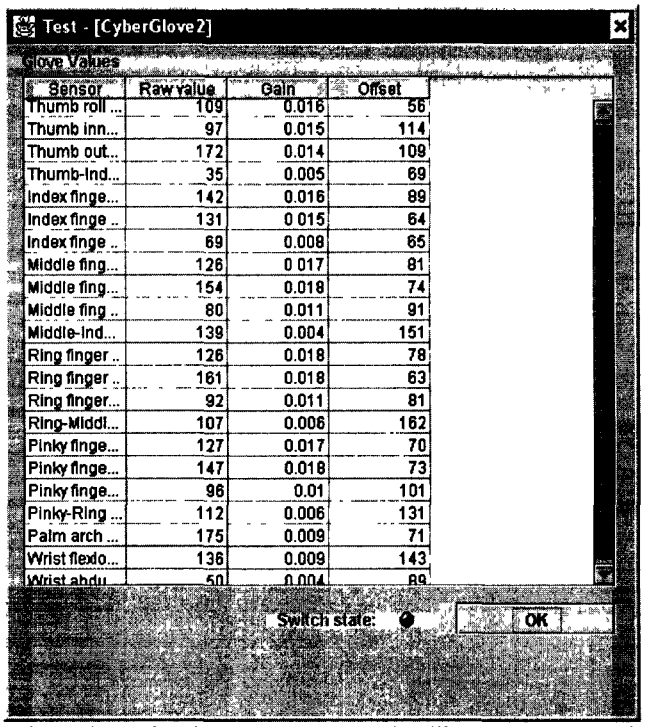

Figure 56: Data captured from the CyberGlove Virtual Hand calibration software for Circular Palm gesture

## Perpendicular Wrist

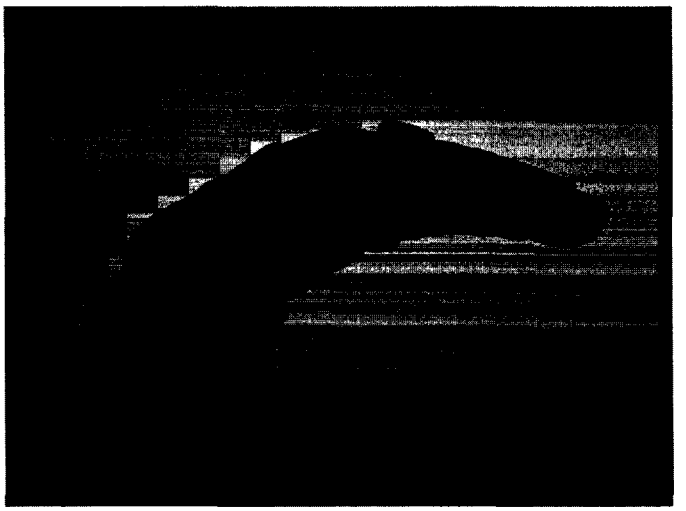
Figure 57: Right Lateral View of Perpendicular Wrist gesture

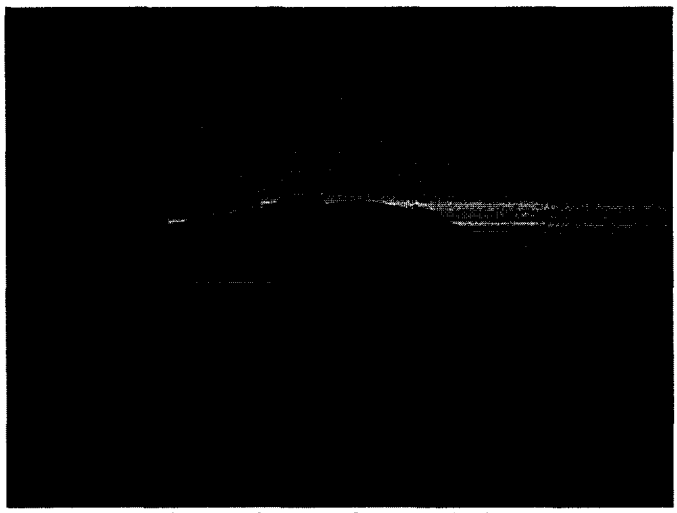

Figure 58: Left Lateral View of Perpendicular Wrist gesture

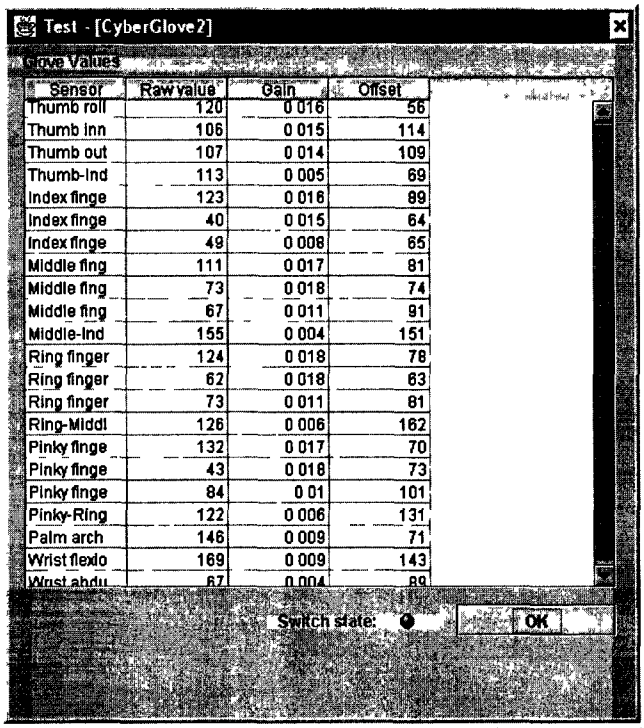


Figure 59: Data captured from the CyberGlove Virtual Hand calibration software for Perpendicular Wrist gesture

## Straight Wrist

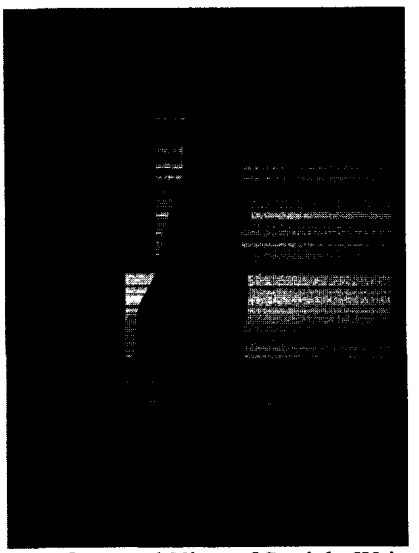

Figure 60: Left Lateral View of Straight Wrist gesture

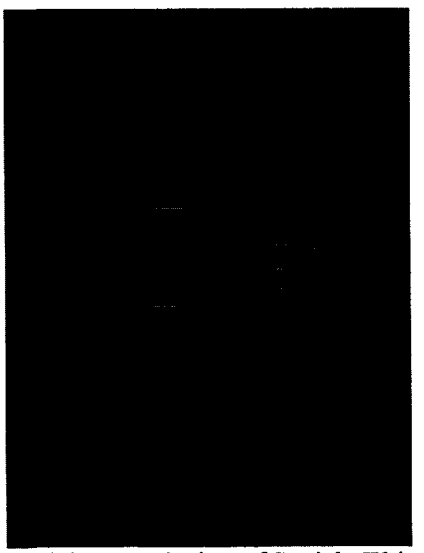

Figure 61: Right Lateral View of Straight Wrist gesture

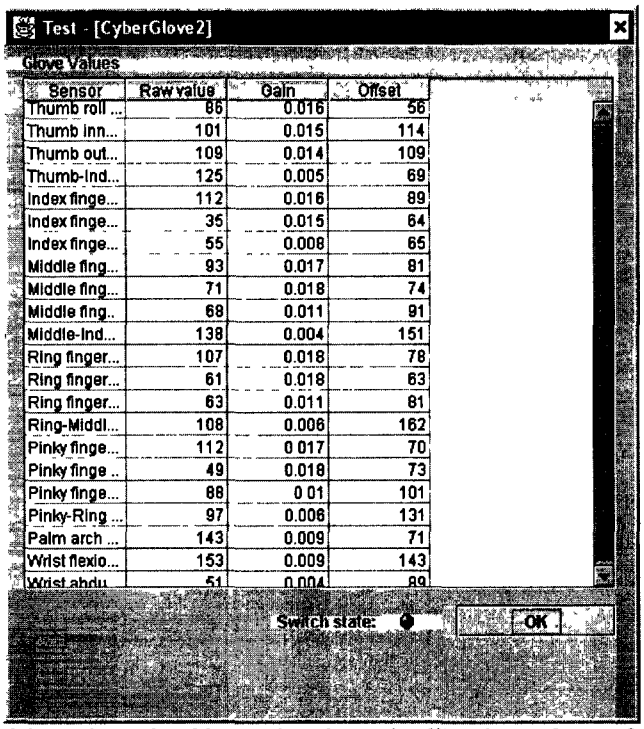

Figure 62: Data captured from the CyberGlove Virtual Hand calibration software for Straight Wrist gesture

## Two Fingers


Figure 63: Right Lateral View of Three Fingers gesture

Figure 64: Left Lateral View of Three Fingers gesture

Figure 65: Dorsal View of Three Fingers gesture

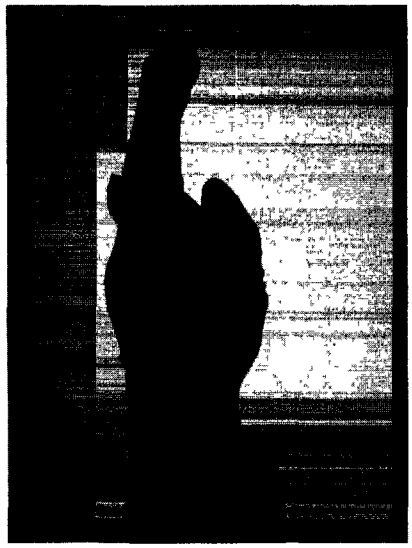

Figure 66: Palmar View of Three Fingers gesture

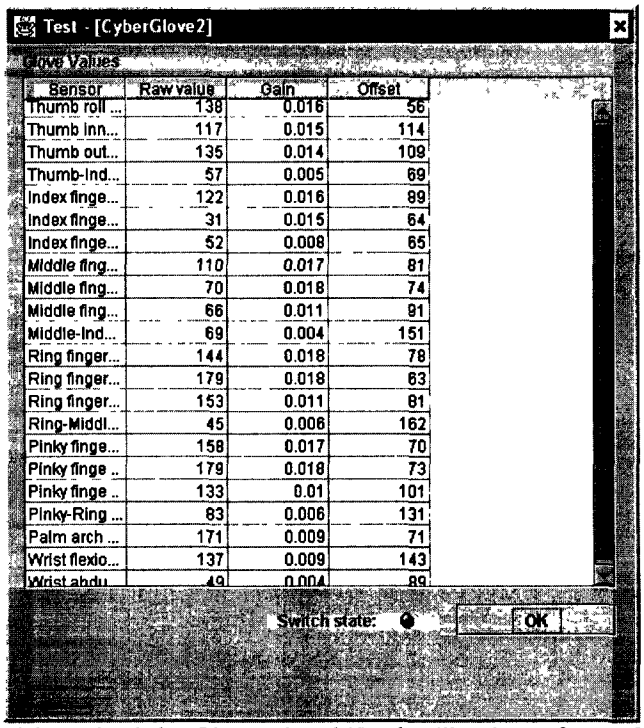


Figure 67: Data captured from the CyberGlove Virtual Hand calibration software for Two Fingers gesture

## Three Fingers

Figure 68: Right Lateral View of Three Fingers gesture

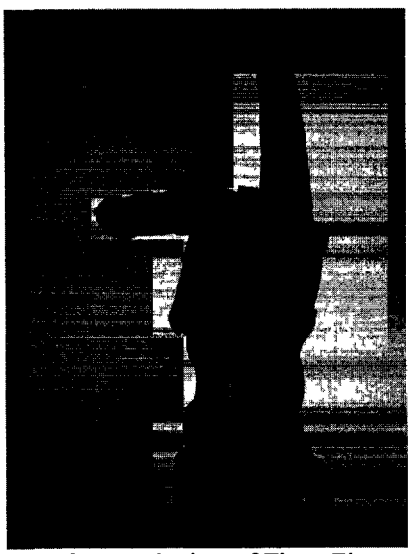

Figure 69: Left Lateral View of Three Fingers gesture

Figure 70: Dorsal View of Three Fingers gesture

Figure 71: Palmar View of Three Fingers gesture

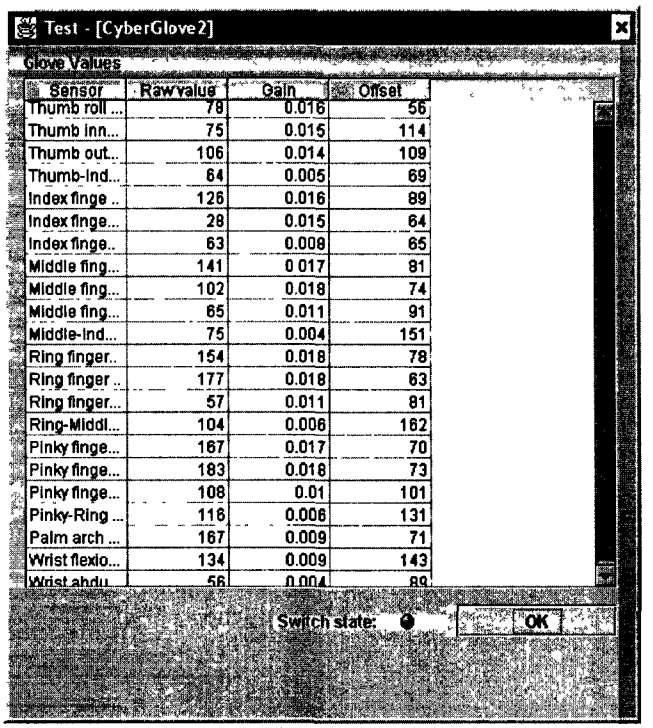

Figure 72: Data captured from the CyberGlove Virtual Hand calibration software for Three Fingers gesture

## Sample Data for Calibration Step 1 – Volunteer #1

| Sen. 1 – Thumb Roll | | Sen. 2 – Thumb inner | | Sen. 3 – Thumb Outer | | Sen. 4 – Thumb Abd. | |
|---|---|---|---|---|---|---|---|
| Angles | Readings | Angles | Readings | Angles | Readings | Angles | Readings |
| 180 | 79 | 180 | 104 | 180 | 105 | 0 | 133 |
| 180 | 86 | 180 | 64 | 180 | 105 | 40 | 69 |
| 180 | 90 | 180 | 70 | 180 | 100 | 45 | 70 |
| 180 | 82 | 180 | 113 | 140 | 138 | 10 | 90 |
| 180 | 80 | 180 | 69 | 180 | 101 | 40 | 59 |
| 180 | 77 | 180 | 129 | 180 | 99 | 0 | 131 |
| 100 | 109 | 175 | 97 | 110 | 172 | 80 | 35 |
| 160 | 86 | 180 | 101 | 180 | 109 | 0 | 125 |
| 180 | 120 | 180 | 106 | 180 | 107 | 0 | 113 |
| 120 | 138 | 175 | 117 | 160 | 135 | 60 | 57 |
| 170 | 78 | 180 | 75 | 180 | 106 | 20 | 64 |

| Gain | 1.65671 | Gain | 1.7959 | Gain | 1.36573 | Gain | 0.179139 |
|---|---|---|---|---|---|---|---|
| Offset | 10.1702 | Offset | 8.48079 | Offset | 9.63294 | Offset | 11.41220 |

| Sen. 5 – Index Inner | | Sen. 6 – Index Middle | | Sen. 7 – Index Outer | | Sen. 8 – Middle Inner | |
|---|---|---|---|---|---|---|---|
| Angles | Readings | Angles | Readings | Angles | Readings | Angles | Readings |
| 180 | 118 | 180 | 35 | 180 | 49 | 180 | 99 |
| 180 | 105 | 180 | 39 | 180 | 48 | 180 | 86 |
| 180 | 119 | 180 | 34 | 180 | 63 | 90 | 139 |
| 135 | 147 | 90 | 153 | 140 | 95 | 135 | 140 |
| 135 | 151 | 90 | 157 | 140 | 123 | 135 | 142 |
| 90 | 153 | 170 | 64 | 180 | 56 | 90 | 150 |
| 150 | 142 | 90 | 131 | 170 | 69 | 150 | 126 |
| 170 | 112 | 180 | 35 | 180 | 55 | 170 | 93 |
| 180 | 123 | 180 | 40 | 180 | 49 | 180 | 111 |
| 180 | 122 | 180 | 31 | 180 | 52 | 170 | 110 |
| 180 | 126 | 180 | 28 | 180 | 63 | 110 | 141 |

| Gain | 1.20026 | Gain | 1.21010 | Gain | 2.28211 | Gain | 1.11140 |
|---|---|---|---|---|---|---|---|
| Offset | 5.27518 | Offset | 72.36870 | Offset | 22.029 | Offset | 9.46030 |

| Sen. 9 – Middle Mid | | Sen. 10 – Middle Outer | | Sen. 11 – Index Abd | | Sen. 12 – Ring Inner | |
|---|---|---|---|---|---|---|---|
| Angles | Readings | Angles | Readings | Angles | Readings | Angles | Readings |
| 180 | 71 | 180 | 72 | 0 | 146 | 180 | 111 |
| 180 | 71 | 180 | 67 | 25 | 113 | 180 | 102 |
| 90 | 174 | 180 | 114 | 0 | 57 | 90 | 149 |
| 90 | 177 | 140 | 93 | 0 | 149 | 135 | 148 |
| 90 | 180 | 140 | 122 | 0 | 128 | 135 | 148 |
| 170 | 90 | 180 | 72 | 0 | 154 | 90 | 152 |
| 90 | 154 | 170 | 80 | 0 | 139 | 150 | 126 |
| 180 | 71 | 180 | 68 | 0 | 138 | 170 | 107 |
| 180 | 73 | 180 | 67 | 0 | 155 | 180 | 124 |
| 180 | 70 | 180 | 66 | 30 | 69 | 160 | 144 |
| 170 | 102 | 180 | 65 | 0 | 75 | 100 | 154 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Gain | 0.98015 | Gain | 1.98834 | Gain | 0.02839 | Gain | 1.02135 |
| Offset | 35.58840 | Offset | 11.66630 | Offset | 1.58528 | Offset | 6.70166 |

| Sen. 13 – Ring Mid | | Sen. 14 – Ring Outer | | Sen. 15 – Middle Abd | | Sen. 16 – Pinky Inner | |
|---|---|---|---|---|---|---|---|
| Angles | Readings | Angles | Readings | Angles | Readings | Angles | Readings |
| 180 | 61 | 180 | 58 | 0 | 109 | 180 | 117 |
| 180 | 64 | 180 | 55 | 15 | 95 | 180 | 102 |
| 90 | 189 | 180 | 120 | 0 | 91 | 90 | 162 |
| 90 | 186 | 140 | 121 | 0 | 103 | 135 | 162 |
| 90 | 188 | 140 | 131 | 0 | 89 | 135 | 168 |
| 170 | 84 | 180 | 63 | 0 | 121 | 90 | 159 |
| 90 | 161 | 170 | 92 | 0 | 107 | 150 | 127 |
| 180 | 61 | 180 | 63 | 0 | 108 | 180 | 112 |
| 180 | 62 | 180 | 73 | 0 | 126 | 180 | 132 |
| 90 | 179 | 110 | 153 | 0 | 45 | 150 | 158 |
| 90 | 177 | 180 | 57 | 0 | 104 | 100 | 167 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Gain | 0.71816 | Gain | 1.54258 | Gain | 0.0124794 | Gain | 0.944087 |
| Offset | 37.81480 | Offset | 27.18300 | Offset | 0.117963 | Offset | 8.32356 |

| Sen. 17 – Pinky Mid | | Sen. 18 – Pinky Outer | | Sen. 19 – Ring Abd | | Sen. 20 – Palm Arch | |
|---|---|---|---|---|---|---|---|
| Angles | Readings | Angles | Readings | Angles | Readings | Angles | Readings |
| 180 | 47 | 180 | 87 | 0 | 103 | 180 | 146 |
| 180 | 44 | 180 | 89 | 20 | 55 | 180 | 141 |
| 90 | 174 | 180 | 120 | 0 | 115 | 180 | 188 |
| 90 | 179 | 140 | 133 | 0 | 109 | 180 | 156 |
| 90 | 175 | 140 | 135 | 0 | 113 | 180 | 175 |
| 180 | 59 | 180 | 81 | 0 | 125 | 180 | 155 |
| 100 | 147 | 170 | 96 | 0 | 112 | 170 | 175 |
| 180 | 49 | 180 | 88 | 0 | 97 | 180 | 143 |
| 180 | 43 | 180 | 84 | 0 | 122 | 180 | 146 |
| 90 | 179 | 120 | 133 | 0 | 83 | 170 | 171 |
| 90 | 183 | 170 | 108 | 0 | 116 | 180 | 167 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Gain | 0.720786 | Gain | 1.48711 | Gain | 0.00884842 | Gain | 1.10115 |
| Offset | 48.0104 | Offset | 9.44343 | Offset | 0.89312 | Offset | 1.69723 |

| Sen. 21 – Wrist Abd | | Sen. 22 – Wrist Flexion | |
|---|---|---|---|
| Angles | Readings | Angles | Readings |
| 180 | 46 | 180 | 156 |
| 180 | 46 | 180 | 162 |
| 180 | 63 | 180 | 132 |
| 180 | 54 | 180 | 136 |
| 180 | 71 | 180 | 135 |
| 180 | 52 | 180 | 143 |
| 180 | 50 | 180 | 136 |
| 110 | 51 | 180 | 153 |
| 180 | 67 | 180 | 169 |
| 180 | 49 | 180 | 137 |
| 180 | 56 | 180 | 134 |

| | | | |
|---|---|---|---|
| Gain | 3.09894 | Gain | 1.23401 |
| Offset | 3.19446 | Offset | 1.29337 |

Table 4: Sample Data and LSR Analysis for Calibration Step 1 for Volunteer #1

## Calibration - Part 2

### Thumb-Index Abduction Sensor

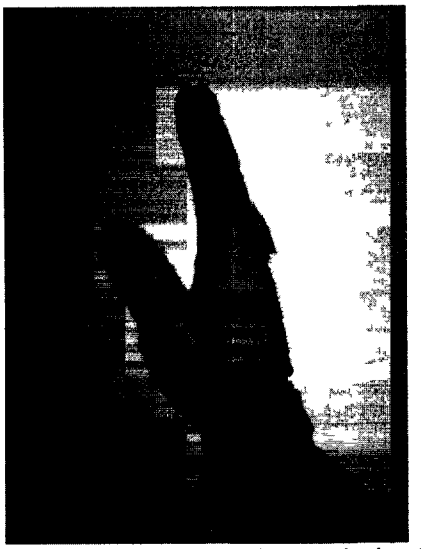

Figure 73: Moving the Left Adjacent Flexion Sensor

Figure 74. Moving the Right Adjacent Flexion Sensor

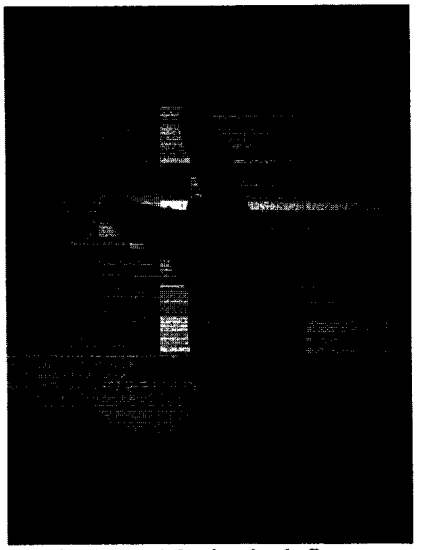

Figure 75: Moving both fingers

## Thumb-Index Abduction Sensor – Sample Data Values for Volunteer #1

**Sensor 4: Thumb-Index Abduction Sensor**
**INDEX FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | INDEX SENSOR | THUMB SENSOR | THUMB ROLL |
|---|---|---|---|---|---|
| 1 | 88 | 20 | 150 | 92 | 99 |
| 2 | 102 | 20 | 150 | 95 | 105 |
| 3 | 96 | 20 | 154 | 93 | 103 |
| 4 | 93 | 20 | 153 | 93 | 101 |
| 5 | 95 | 20 | 160 | 83 | 109 |
| 6 | 85 | 20 | 159 | 86 | 106 |
| 7 | 83 | 20 | 158 | 86 | 106 |
| 8 | 80 | 20 | 157 | 85 | 105 |
| 9 | 78 | 20 | 157 | 85 | 104 |
| 10 | 81 | 20 | 162 | 84 | 103 |

**THUMB FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | INDEX SENSOR | THUMB SENSOR | THUMB ROLL |
|---|---|---|---|---|---|
| 1 | 68 | -10 | 104 | 91 | 141 |
| 2 | 60 | -10 | 106 | 90 | 143 |
| 3 | 66 | -10 | 108 | 91 | 143 |
| 4 | 69 | -10 | 106 | 89 | 135 |
| 5 | 67 | -10 | 106 | 88 | 141 |
| 6 | 71 | -10 | 107 | 93 | 139 |
| 7 | 71 | -10 | 105 | 92 | 138 |
| 8 | 76 | -10 | 106 | 92 | 131 |
| 9 | 70 | -10 | 105 | 92 | 144 |
| 10 | 70 | -10 | 105 | 92 | 144 |

**BOTH FINGERS**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | INDEX SENSOR | THUMB SENSOR | THUMB ROLL |
|---|---|---|---|---|---|
| 1 | 107 | -60 | 164 | 126 | 147 |
| 2 | 106 | -60 | 163 | 124 | 145 |
| 3 | 111 | -60 | 162 | 124 | 146 |
| 4 | 116 | -60 | 155 | 122 | 137 |
| 5 | 116 | -60 | 156 | 122 | 137 |
| 6 | 115 | -60 | 157 | 122 | 139 |
| 7 | 112 | -60 | 157 | 120 | 136 |
| 8 | 109 | -60 | 159 | 119 | 138 |
| 9 | 107 | -60 | 159 | 120 | 138 |
| 10 | 109 | -60 | 158 | 119 | 139 |

| | |
|---|---|
| **CROSS PARAMETER - LEFT** | 0.370702 |
| **CROSS PARAMETER - RIGHT** | 0.0459669 |
| **CROSS PARAMETER - B** | 0.0132987 |

Table 5: Sensor readings and angle measurements for the Thumb-Index abduction angle

## Index-Middle Abduction Sensor

Figure 76: Moving the Left Adjacent Flexion Sensor

Figure 77. Moving the Right Adjacent Flexion Sensor

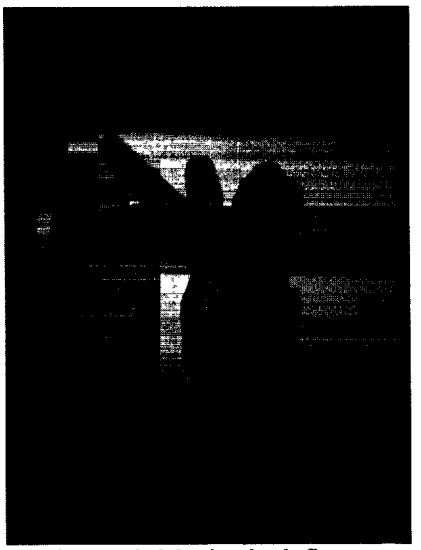

Figure 78: Moving both fingers

## Index-Middle Abduction Sensor – Sample Data Values for Volunteer #1

**Sensor 11: Index-Middle Abduction Sensor**
**MIDDLE FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | MIDDLE SENSOR | INDEX SENSOR |
|---|---|---|---|---|
| 1 | 92 | -10 | 113 | 154 |
| 2 | 94 | -10 | 114 | 154 |
| 3 | 99 | -10 | 112 | 152 |
| 4 | 102 | -10 | 112 | 149 |
| 5 | 103 | -10 | 112 | 148 |
| 6 | 106 | -10 | 110 | 145 |
| 7 | 107 | -10 | 115 | 149 |
| 8 | 97 | -10 | 104 | 141 |
| 9 | 102 | -10 | 109 | 144 |
| 10 | 100 | -10 | 117 | 154 |

**INDEX FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | MIDDLE SENSOR | INDEX SENSOR |
|---|---|---|---|---|
| 1 | 106 | 5 | 144 | 119 |
| 2 | 105 | 5 | 143 | 120 |
| 3 | 104 | 5 | 143 | 119 |
| 4 | 103 | 5 | 143 | 119 |
| 5 | 75 | 5 | 148 | 120 |
| 6 | 87 | 5 | 149 | 123 |
| 7 | 96 | 5 | 149 | 124 |
| 8 | 92 | 5 | 150 | 123 |
| 9 | 93 | 5 | 150 | 123 |
| 10 | 96 | 5 | 151 | 126 |

**BOTH FINGERS**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | MIDDLE SENSOR | INDEX SENSOR |
|---|---|---|---|---|
| 1 | 140 | 0 | 154 | 162 |
| 2 | 143 | 0 | 160 | 164 |
| 3 | 145 | 0 | 152 | 156 |
| 4 | 145 | 0 | 155 | 163 |
| 5 | 144 | 0 | 154 | 158 |
| 6 | 147 | 0 | 156 | 159 |
| 7 | 148 | 0 | 156 | 158 |
| 8 | 150 | 0 | 158 | 165 |
| 9 | 158 | 0 | 154 | 157 |
| 10 | 144 | 0 | 155 | 161 |

| | |
|---|---|
| **CROSS PARAMETER - LEFT** | -0.0057482 |
| **CROSS PARAMETER - RIGHT** | 0.128953 |
| **CROSS PARAMETER - B** | 0.00723572 |

Table 1: Sensor readings and angle measurements for the Index-Middle abduction angle

## Middle-Ring Abduction Sensor

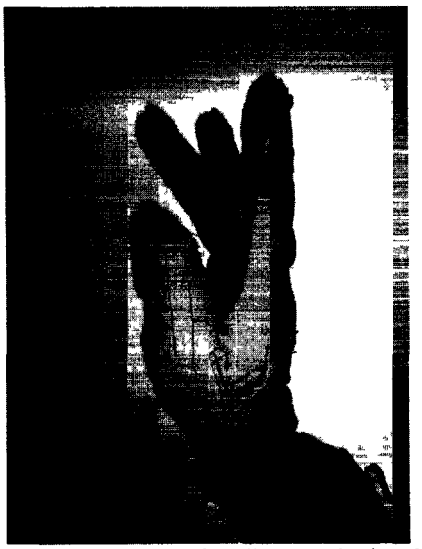

Figure 1: Moving the Left Adjacent Flexion Sensor

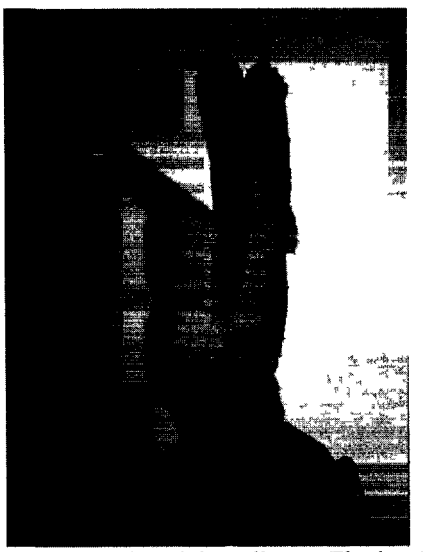

Figure 2: Moving the Right Adjacent Flexion Sensor

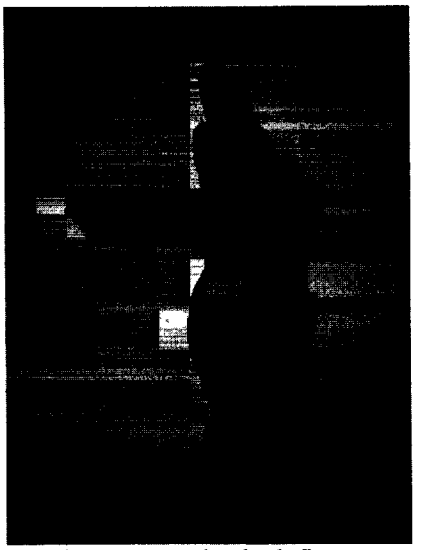

Figure 3: Moving both fingers

## Middle-Ring Abduction Sensor – Sample Data Values for *Volunteer #1*

**Sensor 15: Middle-Ring Abduction Sensor**
**RING FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | RING SENSOR | MIDDLE SENSOR |
|---|---|---|---|---|
| 1 | 98 | -5 | 142 | 112 |
| 2 | 99 | -5 | 141 | 112 |
| 3 | 103 | -5 | 141 | 112 |
| 4 | 96 | -5 | 146 | 116 |
| 5 | 98 | -5 | 143 | 110 |
| 6 | 99 | -5 | 143 | 108 |
| 7 | 100 | -5 | 142 | 109 |
| 8 | 97 | -5 | 137 | 103 |
| 9 | 91 | -5 | 138 | 103 |
| 10 | 95 | -5 | 137 | 101 |

**MIDDLE FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | RIGHT SENSOR | MIDDLE SENSOR |
|---|---|---|---|---|
| 1 | 103 | 0 | 120 | 131 |
| 2 | 99 | 0 | 125 | 137 |
| 3 | 100 | 0 | 119 | 135 |
| 4 | 97 | 0 | 119 | 128 |
| 5 | 98 | 0 | 119 | 128 |
| 6 | 99 | 0 | 120 | 130 |
| 7 | 103 | 0 | 120 | 128 |
| 8 | 100 | 0 | 121 | 129 |
| 9 | 97 | 0 | 121 | 137 |
| 10 | 103 | 0 | 123 | 137 |

**BOTH FINGERS**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | RING SENSOR | MIDDLE SENSOR |
|---|---|---|---|---|
| 1 | 127 | 0 | 150 | 148 |
| 2 | 123 | 0 | 149 | 148 |
| 3 | 123 | 0 | 149 | 149 |
| 4 | 122 | 0 | 149 | 147 |
| 5 | 122 | 0 | 148 | 146 |
| 6 | 121 | 0 | 149 | 150 |
| 7 | 122 | 0 | 148 | 147 |
| 8 | 124 | 0 | 149 | 147 |
| 9 | 126 | 0 | 151 | 151 |
| 10 | 124 | 0 | 151 | 150 |

| | |
|---|---|
| **CROSS PARAMETER - LEFT** | 0.0103298 |
| **CROSS PARAMETER - RIGHT** | 0.0449206 |
| **CROSS PARAMETER - B** | 0.00163526 |

Table 2: Sensor readings and angle measurements for the Middle-Ring abduction angle

## Ring-Pinky Abduction Sensor

Figure 4: Moving the Left Adjacent Flexion Sensor

Figure 5: Moving the Right Adjacent Flexion Sensor

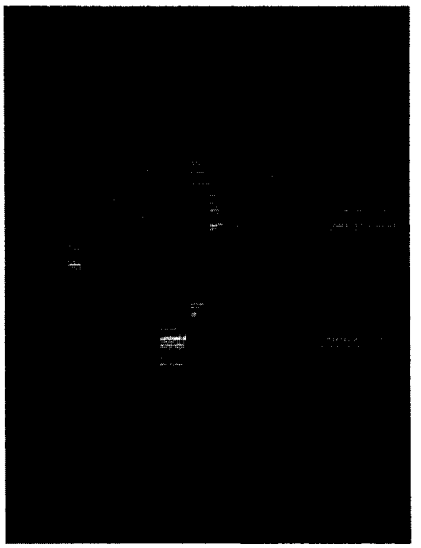

Figure 6: Moving both fingers

## Ring-Pinky Abduction Sensor – Sample Data Values for Volunteer #1

**Sensor 19: Ring-Pinky Abduction Sensor**
**PINKY FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | PINKY SENSOR | RING SENSOR |
|---|---|---|---|---|
| 1 | 83 | 10 | 123 | 141 |
| 2 | 81 | 10 | 122 | 140 |
| 3 | 80 | 10 | 122 | 141 |
| 4 | 79 | 10 | 122 | 141 |
| 5 | 76 | 10 | 123 | 148 |
| 6 | 75 | 10 | 123 | 146 |
| 7 | 77 | 10 | 124 | 146 |
| 8 | 76 | 10 | 123 | 146 |
| 9 | 78 | 10 | 122 | 141 |
| 10 | 77 | 10 | 122 | 142 |

**RING FINGER ONLY**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | PINKY SENSOR | RING SENSOR |
|---|---|---|---|---|
| 1 | 88 | 0 | 149 | 121 |
| 2 | 87 | 0 | 147 | 120 |
| 3 | 89 | 0 | 149 | 122 |
| 4 | 90 | 0 | 150 | 123 |
| 5 | 90 | 0 | 149 | 124 |
| 6 | 83 | 0 | 146 | 119 |
| 7 | 85 | 0 | 147 | 120 |
| 8 | 87 | 0 | 147 | 120 |
| 9 | 76 | 0 | 152 | 123 |
| 10 | 79 | 0 | 147 | 119 |

**BOTH FINGERS**

| TRIAL # | ABD SENSOR | MEASURED ANGLE | PINKY SENSOR | RING SENSOR |
|---|---|---|---|---|
| 1 | 86 | 0 | 156 | 152 |
| 2 | 93 | 0 | 160 | 153 |
| 3 | 97 | 0 | 159 | 154 |
| 4 | 97 | 0 | 158 | 152 |
| 5 | 94 | 0 | 159 | 151 |
| 6 | 94 | 0 | 159 | 151 |
| 7 | 100 | 0 | 162 | 156 |
| 8 | 98 | 0 | 159 | 152 |
| 9 | 97 | 0 | 160 | 152 |
| 10 | 100 | 0 | 160 | 153 |

| | |
|---|---|
| **CROSS PARAMETER - LEFT** | 0.0136137 |
| **CROSS PARAMETER - RIGHT** | -0.0686356 |
| **CROSS PARAMETER - B** | -0.000024522 |

Table 3: Sensor readings and angle measurements for the Ring-Pinky abduction angle

# Appendix B – Matlab Module for LSR Analysis

## B-1. LSR Analyzer

```
outputFile = fopen('outputfile.txt', 'w');
%sensor1
filename = 'theta1.txt';
A = textread(filename, '%d')
sensorFile = 'sensor1.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'1 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor2
filename = 'theta2.txt';
A = textread(filename, '%d')
sensorFile = 'sensor2.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'2 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor3
filename = 'theta3.txt';
A = textread(filename, '%d')
sensorFile = 'sensor3.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'3 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor4
%Step 1: Calculating the Gain and Offset
filename = 'theta4.txt';
theta = textread(filename, '%d')
sensorFile = 'sensor4.txt';
sensorValues = textread(sensorFile, '%d')
[g, bint, r, rint] = regress(theta, sensorValues, 0.05)
offset = mean(r)

%Step 2: Calculating the Cross Parameters
%Part 1.a - Calculating delta theta R
realThetaFileR = 'realTheta4R.txt';
realThetaR = textread(realThetaFileR, '%d')
sensorFileR = 'sensor4R.txt';
sensorValuesR = textread(sensorFileR, '%d')
calculatedThetaR = g * sensorValuesR + offset

deltaThetaR = calculatedThetaR - realThetaR

%Part 2 - Calculating Kr and b1
```

```
%The equation is deltaTheta = kl xl + kr xr + b
%for kl = 0, deltaTheta = kr xr + b
rightSensorFile = 'sensorR4.txt';
rightSensorValues = textread(rightSensorFile, '%d')
[kr, krint, b1, b1int] = regress (deltaThetaR, rightSensorValues, 0.05)

%Part 1.b - Calculating delta theta L
realThetaFileL = 'realTheta4L.txt';
realThetaL = textread(realThetaFileL, '%d')
sensorFileL = 'sensor4L.txt';
sensorValuesL = textread(sensorFileL, '%d')
calculatedThetaL = g * sensorValuesL + offset

deltaThetaL = calculatedThetaL - realThetaL

%Part 3 - Calculating Kl and b2
%for kr = 0, deltaTheta = kl xl + b
leftSensorFile = 'sensorL4.txt';
leftSensorValues = textread(leftSensorFile, '%d')
[kl, klint, b2, b2int] = regress (deltaThetaL, leftSensorValues, 0.05)

%Part 4 - Calculating b
%b2 and b1 should be close in value.
bVal = [b1, b2]
b = mean (bVal)
b = mean (b)

%Step 3: Output all results
fprintf(outputFile,'4 %G %G %G %G %G\n', g, offset, kl, kr, b )

%sensor5
filename = 'theta5.txt';
A = textread(filename, '%d')
sensorFile = 'sensor5.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'5 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor6
filename = 'theta6.txt';
A = textread(filename, '%d')
sensorFile = 'sensor6.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'6 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor7
filename = 'theta7.txt';
A = textread(filename, '%d')
sensorFile = 'sensor7.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
```

```
fprintf(outputFile,'7 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor8
filename = 'theta8.txt';
A = textread(filename, '%d')
sensorFile = 'sensor8.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'8 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor9
filename = 'theta9.txt';
A = textread(filename, '%d')
sensorFile = 'sensor9.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'9 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor10
filename = 'theta10.txt';
A = textread(filename, '%d')
sensorFile = 'sensor10.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'10 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor11
%Step 1: Calculating the Gain and Offset
filename = 'theta11.txt';
theta = textread(filename, '%d')
sensorFile = 'sensor11.txt';
sensorValues = textread(sensorFile, '%d')
[g, bint, r, rint] = regress(theta, sensorValues, 0.05)
offset = mean(r)

%Step 2: Calculating the Cross Parameters
%Part 1.a - Calculating delta theta R
realThetaFileR = 'realTheta11R.txt';
realThetaR = textread(realThetaFileR, '%d')
sensorFileR = 'sensor11R.txt';
sensorValuesR = textread(sensorFileR, '%d')
calculatedThetaR = g * sensorValuesR + offset

deltaThetaR = calculatedThetaR - realThetaR

%Part 2 - Calculating Kr and b1
%The equation is deltaTheta = k1 x1 + kr xr + b
%for k1 = 0, deltaTheta = kr xr + b
rightSensorFile = 'sensorR11.txt';
rightSensorValues = textread(rightSensorFile, '%d')
[kr, krint, b1, b1int] = regress (deltaThetaR, rightSensorValues, 0.05)
```

```
%Part 1.b - Calculating delta theta L
realThetaFileL = 'realTheta11L.txt';
realThetaL = textread(realThetaFileL, '%d')
sensorFileL = 'sensor11L.txt';
sensorValuesL = textread(sensorFileL, '%d')
calculatedThetaL = g * sensorValuesL + offset

deltaThetaL = calculatedThetaL - realThetaL

%Part 3 - Calculating K1 and b2
%for kr = 0, deltaTheta = k1 x1 + b
leftSensorFile = 'sensorL11.txt';
leftSensorValues = textread(leftSensorFile, '%d')
[k1, klint, b2, b2int] = regress (deltaThetaL, leftSensorValues, 0.05)

%Part 4 - Calculating b
%b2 and b1 should be close in value.
bVal = [b1, b2]
b = mean (bVal)
b = mean (b)

%Step 3: Output all results
fprintf(outputFile,'11 %G %G %G %G %G\n', g, offset, k1, kr, b )

%sensor12
filename = 'theta12.txt';
A = textread(filename, '%d')
sensorFile = 'sensor12.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'12 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor13
filename = 'theta13.txt';
A = textread(filename, '%d')
sensorFile = 'sensor13.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'13 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor14
filename = 'theta14.txt';
A = textread(filename, '%d')
sensorFile = 'sensor14.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'14 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor15
%Step 1: Calculating the Gain and Offset
filename = 'theta15.txt';
theta = textread(filename, '%d')
```

```
sensorFile = 'sensor15.txt';
sensorValues = textread(sensorFile, '%d')
[g, bint, r, rint] = regress(theta, sensorValues, 0.05)
offset = mean(r)

%Step 2: Calculating the Cross Parameters
%Part 1.a - Calculating delta theta R
realThetaFileR = 'realTheta15R.txt';
realThetaR = textread(realThetaFileR, '%d')
sensorFileR = 'sensor15R.txt';
sensorValuesR = textread(sensorFileR, '%d')
calculatedThetaR = g * sensorValuesR + offset

deltaThetaR = calculatedThetaR - realThetaR

%Part 2 - Calculating Kr and b1
%The equation is deltaTheta = k1 x1 + kr xr + b
%for k1 = 0, deltaTheta = kr xr + b
rightSensorFile = 'sensorR15.txt';
rightSensorValues = textread(rightSensorFile, '%d')
[kr, krint, b1, b1int] = regress (deltaThetaR, rightSensorValues, 0.05)

%Part 1.b - Calculating delta theta L
realThetaFileL = 'realTheta15L.txt';
realThetaL = textread(realThetaFileL, '%d')
sensorFileL = 'sensor15L.txt';
sensorValuesL = textread(sensorFileL, '%d')
calculatedThetaL = g * sensorValuesL + offset

deltaThetaL = calculatedThetaL - realThetaL

%Part 3 - Calculating K1 and b2
%for kr = 0, deltaTheta = k1 x1 + b
leftSensorFile = 'sensorL15.txt';
leftSensorValues = textread(leftSensorFile, '%d')
[k1, k1int, b2, b2int] = regress (deltaThetaL, leftSensorValues, 0.05)

%Part 4 - Calculating b
%b2 and b1 should be close in value.
bVal = [b1, b2]
b = mean (bVal)
b = mean (b)

%Step 3: Output all results
fprintf(outputFile,'15 %G %G %G %G %G\n', g, offset, k1, kr, b )

%sensor16
filename = 'theta16.txt';
A = textread(filename, '%d')
sensorFile = 'sensor16.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'16 %G %G %G %G %G\n', b, avr, 0, 0, 0 )
```

```
%sensor17
filename = 'theta17.txt';
A = textread(filename, '%d')
sensorFile = 'sensor17.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'17 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor18
filename = 'theta18.txt';
A = textread(filename, '%d')
sensorFile = 'sensor18.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'18 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor19
%Step 1: Calculating the Gain and Offset
filename = 'theta19.txt';
theta = textread(filename, '%d')
sensorFile = 'sensor19.txt';
sensorValues = textread(sensorFile, '%d')
[g, bint, r, rint] = regress(theta, sensorValues, 0.05)
offset = mean(r)

%Step 2: Calculating the Cross Parameters
%Part 1.a - Calculating delta theta R
realThetaFileR = 'realTheta19R.txt';
realThetaR = textread(realThetaFileR, '%d')
sensorFileR = 'sensor19R.txt';
sensorValuesR = textread(sensorFileR, '%d')
calculatedThetaR = g * sensorValuesR + offset

deltaThetaR = calculatedThetaR - realThetaR

%Part 2 - Calculating Kr and b1
%The equation is deltaTheta = kl xl + kr xr + b
%for kl = 0, deltaTheta = kr xr + b
rightSensorFile = 'sensorR19.txt';
rightSensorValues = textread(rightSensorFile, '%d')
[kr, krint, b1, b1int] = regress (deltaThetaR, rightSensorValues, 0.05)

%Part 1.b - Calculating delta theta L
realThetaFileL = 'realTheta19L.txt';
realThetaL = textread(realThetaFileL, '%d')
sensorFileL = 'sensor19L.txt';
sensorValuesL = textread(sensorFileL, '%d')
calculatedThetaL = g * sensorValuesL + offset

deltaThetaL = calculatedThetaL - realThetaL

%Part 3 - Calculating Kl and b2
%for kr = 0, deltaTheta = kl xl + b
```

```
leftSensorFile = 'sensorL19.txt';
leftSensorValues = textread(leftSensorFile, '%d')
[kl, klint, b2, b2int] = regress (deltaThetaL, leftSensorValues, 0.05)

%Part 4 - Calculating b
%b2 and b1 should be close in value.
bVal = [b1, b2]
b = mean (bVal)
b = mean (b)

%Step 3: Output all results
fprintf(outputFile,'19 %G %G %G %G %G\n', g, offset, kl, kr, b )

%sensor20
filename = 'theta20.txt';
A = textread(filename, '%d')
sensorFile = 'sensor20.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'20 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor21
filename = 'theta21.txt';
A = textread(filename, '%d')
sensorFile = 'sensor21.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'21 %G %G %G %G %G\n', b, avr, 0, 0, 0 )

%sensor22
filename = 'theta22.txt';
A = textread(filename, '%d')
sensorFile = 'sensor22.txt';
B = textread(sensorFile, '%d')
[b, bint, r, rint] = regress(A, B, 0.05)
avr = mean(r)
fprintf(outputFile,'22 %G %G %G %G %G\n', b, avr, 0, 0, 0 )
fclose(outputFile)
```