# A Validation of a Simulation Environment for Motion Sensing Electronic Textiles

Christopher Einsmann

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Dr. Thomas L. Martin, Chair
Dr. Mark T. Jones
Dr. Peter M. Athanas

February 10, 2006
Blacksburg, Virginia

Keywords: E-textiles, wearable computing, gait analysis, gyroscopes, accelerometers, simulation

# A Validation of a Simulation Environment for Motion Sensing Electronic Textiles

Christopher Einsmann

(ABSTRACT)

*Electrical components constantly being scaled down in size allows for small, inexpensive sensors to be placed on or around the human body for motion sensing applications. In addition, the merging of textiles with electrical components, known as electronic textiles (e-textiles), allows for these sensors to be placed directly on a wearable fabric. Simulation allows for extensive application testing and verification before prototype development. This thesis presents a simulation environment for motion sensing e-textiles. Specifically, this environment incorporates motion capture position data to simulate a rate sensing gyroscope and a dual-axis accelerometer. In addition, this simulation environment is applied to the field of gait analysis, which is the process of quantification and interpretation of a person's stride, to calculate a subject's step length.*

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

In the early 1990s, Mark Weiser presented the seminal paper in the field of ubiquitous and wearable computing [1]. This vision focused on the idea that the most profound technologies are those that "weave themselves into the fabric of everyday life." As the computer and electronics industries progress, components constantly being scaled down in size allows for small computational devices to be placed on or around the human body, hence "wearable computing." Electronic textiles (e-textiles) are fabrics that have interconnections, sensors, processors and or other computational devices integrated into the cloth. Potential applications for e-textiles and wearable computing include context awareness, motion sensing, and medical monitoring, among others.

The importance of simulating e-textile applications has been suggested in numerous studies by the Virginia Tech E-Textiles Laboratory [2, 3, 4, 5]. The primary goal of this thesis is to validate a simulation environment for a motion sensing e-textile garment and apply that

simulation to a medical monitoring application, specifically gait analysis.

## 1.2 Contributions

This thesis presents the design and implementation of a simulation environment for motion sensing e-textiles. Simulation models for a dual axis accelerometer and a rate sensing gyroscope are generated. The ability to accurately simulate these sensors allows for e-textile applications to be designed and simulated before implementation of a prototype. After describing these models, motion capture data is used to validate the models. Also, this thesis attempts to apply e-textile motion sensing to the field of gait analysis by improving upon the step length results found in [4]. The simulation environment is incorporated to analyze and discuss those results.

## 1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 presents the background information on wearable computing, e-textiles and gait analysis necessary to understand this research. The e-textile prototype used for this research is described in Chapter 3. Chapter 4 discusses the simulation environment for motion sensing e-textiles. An attempt to derive the step length of a subject for gait analysis is described is Chapter 5. Finally, Chapter 6 concludes and discusses future work.

# Chapter 2

# Background

The areas of research related to this thesis are electronic textiles, wearable computing and gait analysis. The first section of this chapter discusses electronic textiles and wearable computing, while the second section presents background information on gait analysis and motion capture systems.

## 2.1 Electronic Textiles and Wearable Computing

The field of e-textiles encompasses both wearable fabrics and large scale non-wearable fabrics, such as those that can be used in acoustic beamforming applications [6]. E-textiles designed for wearable computing applications will be focused on in this thesis. A comprehensive overview of research in the field of e-textiles can be found in [7].

One of the first efforts in this area was the wearable motherboard project at the Georgia Institute of Technology [8]. The wearable motherboard provided for the first time a systematic and unobtrusive method of monitoring human vital signs. This garment incorporated optical

fibers to detect bullet wounds, as its intended application was the ambulatory monitoring of soldiers. The WEALTHY project integrated "strain fabric sensors," made of piezoresistive yarns and fabric electrodes to monitor physiological signs such as respiration, heart and muscle activity, and temperature of a subject [9]. The MIT Media Laboratory presented a number of e-textile applications [10] incorporating textile-based circuitry and sensing. A row and column fabric keyboard constructed from highly conductive metallic organza made music when crushed or crumpled using direct electrical connections and a dress was designed that emitted a display of light as the wearer moved. In addition to these, they created a jacket that played MIDI music, a ball that played music when squeezed and an electronic table cloth. These applications demonstrated the advantages of textile-based circuitry over printed circuitry.

Another class of applications for wearable computing and e-textiles are those that sense a user's context or motion. Context awareness is the ability of a computing system to be cognizant of the user's environment [11]. Context awareness asks four main questions: where is the user, who is the user with, what resources are nearby [12] and what is the user doing [13]. Activity awareness is a subset of context awareness that asks what the user is doing. This encompasses whether the person is walking running, biking, or if idle, what position the person is in: standing, sitting, or lying down. A plethora of sensors are used to create this awareness, including microphones, accelerometers, gyroscopes, temperature sensors, light sensors, video cameras and piezoelectrics among others.

It is important to understand the evolution of raw data from the sensors to context. The first step is data acquisition. This is usually done by having the output of the sensor input to a microprocessor and converting the electrical signal into a digital value in representative units (for example, $g$'s for an accelerometer). In many applications, especially those with multiple sensors, awareness is achieved through different clustering algorithms. Before the

data can be input into one of these learning algorithms, it is helpful to pre-process that data. In literature, this is referred to as "data cooking" [14] or extracting "cues" [15]. The next step is to input this data into a machine learning algorithm. The goal of this step is to cluster different sets of input combinations into groups that identify with a certain location or action. Using these clusters, user context is extracted. This is usually done by hand annotation of the clusters or labeling based on the last input to the algorithms during training.

The goal of using sensors for context awareness is to create a high-level inference from the raw data. If only the raw sensor data was used as inputs to pattern recognition or learning algorithms, noise and other effects would cause the algorithms to not meet their potential [15]. For this reason, simple preprocessing routines are used to extract higher level features from the data. The preprocessing step adds no new knowledge to the application but simply reinterprets the data into more meaningful terms [14]. Depending on the type of sensor, different features can be extracted. Another advantage of extracting features is that it has the potential to separate the sensors from the features, allowing them to be developed and replaced independently of each other [16].

Some examples of applications that use these techniques include [17], [14], and [15]. A wearable sensor badge and sensor jacket was developed in [17] to sense perambulatory activities for context awareness. The prototype "Sensor Badge" was used to classify a person's activity as sitting, standing, lying on stomach, lying on back, walking or running. The "Sensor Jacket" incorporated knitted stretch sensors along with conductive tracking to detect the posture and movements of the wearer. Van Laerhoven applied Kohonen Self-Organizing Maps, a type of neural network, to detect a user's activity. Golding used cheap, wearable sensors along with machine learning techniques for indoor navigation in [14].

The benefits of using piezoelectric films as shape sensors in e-textiles were presented in [4]

and [18]. In the former, a prototype piezoelectric glove that functioned as a virtual keyboard was designed to illustrate the use of piezoelectrics in e-textiles. In the latter, piezoelectric sensors were used to sense the shape of a user's knees.

While the prototypes discussed thus far have been promising, most of them involved hardware prototypes designed specifically for one or two people without the aid of simulation. The importance of simulating e-textile applications was emphasized in [2], [3] and [5]. In [2], a simulation environment that combined models of the physical environment, human locomotion, sensor behavior, network communication, power consumption and software execution was described. A model of a person wearing an e-textile and the effect of faults on the e-textile system was analyzed in [3]. Simulation was utilized for a feasibility study for a self-contained, wearable, time-of-flight motion capture system in [5]. While context awareness is not a direct focus of this thesis, simulating e-textile applications is. A validated simulation of both an accelerometer and a gyroscope allows for a variety of context aware applications to be simulated and generalized across a range of subjects before prototype development.

## 2.2 Gait Analysis and Motion Capture Systems

Motivation for this research is the desire for a more convenient technique for obtaining gait analysis measurements than the currently used motion capture system, and the idea that this is an ideal application for e-textiles and wearable computing. Gait analysis is the process of quantification and interpretation of a person's gait, that is, the way a person walks. Typically, motion capture systems involving multiple cameras are used to record a subject walking. The subject's position data is logged in three dimensional coordinates, and force data is gathered via a force plate. From this data, different characteristics of the subject's gait can be extracted. Common measures extracted during gait analysis include step length,

heel strike velocity, sliding heel velocity, slip distance, velocity of center of mass, and the required coefficient of friction [19]. Advantages of using e-textiles for gait analysis over a motion capture system include lower cost and the mobility of e-textiles. With e-textiles, a subject's gait can be analyzed at home or in the office, instead of requiring the subject to walk in a specialized motion capture lab.

Gait variability was linked to falls in the elderly in [20]. This emphasized a need to differentiate between those with stable gaits and those whose gaits are less stable. Dingwell and Cusumano used accelerometers and electrogoniometers to calculate average maximum Lyapunov exponents to assess dynamic stability in [21]. This same principle was applied to data from an e-textile and compared to the results from motion capture data in [22].

A first pass at applying e-textiles to gait analysis was made in [4] and this thesis attempts to build off of that progress. In [4], accelerometers were placed slightly above the ankle of the test subject in order to collect data while the subject was walking. From this data, the accelerations of the X-axis of the accelerometer over a step were integrated twice to calculate the subject's step length. In order for this method to be accurate, the X-axis of the accelerometer would have to stay parallel to the ground during the entire step. As a subject rotates his or her leg during the back swing of the step, the accelerometer changes orientation. Change in the orientation of the accelerometer with respect to the ground causes a corresponding change on the accelerometer output due to gravity. In this situation, the X-axis of the accelerometer senses accelerations in the Z-axis as well as the X-axis depending on the orientation of the accelerometer throughout the step. In order to correct for this issue and calculate step length accurately, an approximation was developed that attempted to correct for the change in orientation. Because of this change in orientation, the resulting velocity curve had a non-zero final velocity. A corresponding discrepancy was found in the acceleration. Over the interval that had the error, a three point, piecewise linear fit

was applied to the acceleration, which greatly improved the results. This fit improved the resulting final velocity to be closer to the expected zero value. Each calculated step length was compared to the value calculated in Virginia Tech's Locomotion Laboratory. After the fit was applied, an average step length error of 7% was found with this method. However, this approximation had much worse results for some subjects, hence this thesis seeks to prove that a better solution then the approximation is the incorporation of a gyroscope to sense the change in orientation. This also requires the use of the Y axis acceleration.

# Chapter 3

# E-textile Prototype

The research in this thesis was focused on exploring possible applications for e-textiles. This chapter describes the e-textile garment that was used for that purpose. For the work discussed in [4], a prototype pair of e-textile pants was developed. This thesis builds directly on that work, creating a second iteration of those pants. Section 3.1 describes the basic hardware design of the garment, while Section 3.2 discusses improvements made to the communication protocol incorporated in the garment. Section 3.3 describes how the e-textile synchronizes the data from the different devices attached to the garment. Characteristics of the specific accelerometers used for this research are discussed in Sections 3.4 and 3.5.

## 3.1 E-textile Design

The e-textile pants consisted of e-Textile Attached Gadgets (e-TAGS) interconnected to wires woven into the textile [23]. The prototype e-TAGs consisted of accelerometers or piezoelectric film sensors attached to a PIC16LF18 microcontroller. Raw data from each sensor was collected and processed by each corresponding microcontroller. Each device

would communicate with a serial e-TAG over an $I^2C$ interface. The serial e-TAG, with an Atmel AVR Atmega8 microcontroller [24], functioned as the master node in the $I^2C$ protocol and polled each sensor in a "round robin" fashion for its most recently collected data. The serial node was connected to an IPAQ through an RS232 port and relayed this data over an ad-hoc wireless network to a laptop, where further data processing took place.

For the second iteration of this e-textile, a few changes were made. First, each PIC16LF18 microcontroller was replaced with an Atmel Atmega8. This change was made due to the availability of the Atmel AVR toolset, including a C cross-compiler that allows software for these microcontrollers to be written in C instead of assembly. Also, as discussed in Chapter 1, gyroscopes were added to operate in conjunction with the dual-axis accelerometers on each acceleration e-TAG. In the previous application, the piezoelectric strips were used to sense bending of the knee [25, 26]. In this application, a piezoelectric was taped to the bottom of the subject's shoe to detect when the subject's heel strikes the ground [27]. Another addition to the pants was a Bluetooth module that provided the function of the "master" e-TAG as well as that of the IPAQ. The Bluetooth module abstracts away the details of the radio communication and functions on both the pants and the host machine as a simple RS232 port. Another change made to the communication interface of these pants is that the multi-master part of the $I^2C$ protocol was implemented to allow for any e-TAG to take control of the bus, as discussed in Section 3.2.

## 3.2 $I^2C$ and Multi-Master

The communication protocol implemented in the e-textile used for sensor communication in this application was $I^2C$ [28]. This is a two-wire bus, with data and clock lines, that allows many different types of integrated circuits to communicate and share data. Fundamental to

this protocol's operation is that each bus line implements a wired-AND function by having both lines connected to a pull-up resistor, resulting in each line being tied high when no data is being transmitted. Each line is then in a low state when any device outputs a 0 to the bus. Each device on the system has a 7-bit address associated with it. The protocol operates off the notion that one node is a master and any other nodes involved in the communication are slaves. The master can choose to either send or receive data and also is responsible for generating the clock. Figure 3.1 shows a typical data transmission over $I^2C$. The master begins the data transmission process by initiating a START signal, followed by sending the address of a slave along with a final bit that specifies read or write. A START is specified by pulling the data line low while the clock is still high. Data is then transmitted one byte at a time, with the receiver sending an ACK bit upon successfully receiving the byte [24]. An ACK is specified by the slave pulling the data line high in the ninth cycle (after a cycle for each bit in the byte). Data transmission is terminated by the master sending a STOP signal, which is when the master pulls the data line high while the clock is also high.

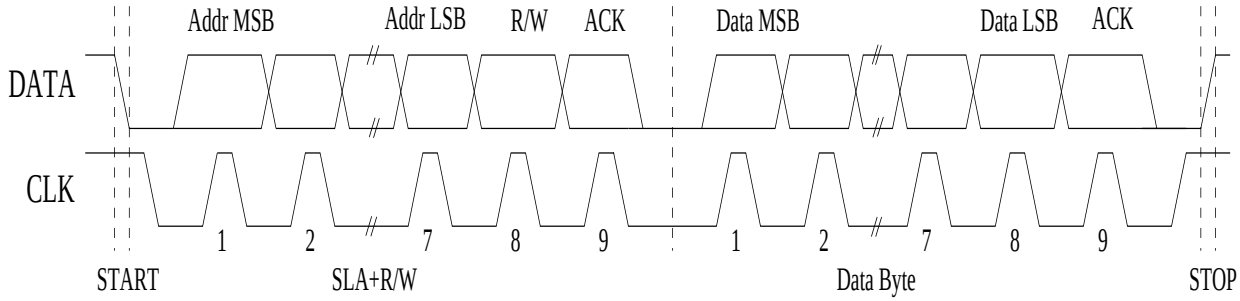


Figure 3.1: Typical $I^2C$ Transmission, based upon a figure in [24]

For the prototype, each sensor's microcontroller was constantly sampling data from its sensor, and when asked by the master e-TAG, would transmit the most recently collected data over $I^2C$. This caused unknown variation in the time between samples, and, at best, an inconsistent sampling rate. When computing discrete integration, the time interval between

samples must be known. Since the data from the sensors was being integrated multiple times, an inconsistent sampling rate was found to be unacceptable. To resolve this issue, two main changes were made. First, the microcontrollers were changed to sample the sensors at a known time interval using the Atmega8's timer interrupt. Each sample was numbered using an 8-bit counter. The second change was that instead of the serial e-TAG requesting data, each individual node would initiate the data transmission to send the data over $I^2C$. Therefore, multiple nodes needed the ability to become master of the bus, where previously only one node could be the master.

In general, the limitation of only having one master e-TAG is an undesirable characteristic for an e-textile. This limits the application to that in which one master button is responsible for the routing of all data. Wearable computing and e-textiles should not be limited by this. One can imagine many applications that require intercommunication between nodes as well communication to a node that transmits data from the e-textile. For example, two nodes could need to access shared memory that is connected to the e-textile.

The $I^2C$ protocol provides multi-master support to overcome this problem. There are two main issues the protocol resolves in the multi-master system. The first is that since the master is responsible for generating a clock pulse, there must be a way to synchronize masters with different clock frequencies. This is built directly into the wired-ANDing of the bus lines. Whenever a low is output to the clock line by any device, it drives the line low. Multiple masters use this to create one clock frequency. This creates one clock from multiple masters' clocks that has the high period of the master with the shortest high period and a low period equal to that of the master with the longest low period [24].

The second issue that the protocol resolves is choosing which master is allowed to complete its transmission when multiple masters simultaneously try to gain control of the bus. Each master is responsible for monitoring both lines to make sure the value it output matches the

value on the line. This is fundamental to $I^2C$'s arbitration process. Each master will monitor the data line, and when the data line does not match the value output by the master, it has lost the arbitration and should immediately go to slave mode. The first master to output a low value while the other(s) output a high value wins the arbitration.

The protocol was implemented in C for the Atmega8's. These microcontrollers have a "Two-wire Serial Interface" built into their hardware, though additional software is necessary for the specific application. The interface built into the microcontroller relies on software response to events. When an $I^2C$ event occurs, the hardware issues an interrupt and a status register is updated with the appropriate event code. In the data sheet for the microcontrollers, after each event on the bus, for each mode (master receiver, master transmitter, slave receiver and slave transmitter) there is a list of software response options, shown in tables 66-69 in [24]. For this application, data loss must be minimized. For this reason, when an unexpected state is entered, instead terminating the transmission, the data transmission is restarted from a known point.

## 3.3 Data Synchronization

In this e-textile prototype, multiple e-TAGs sent and processed data at various time intervals. When synthesizing data from multiple sensors, it is of the utmost importance that the data be synchronized. As mentioned above, the Atmega8's timer interrupt function was used to set the sampling rate. Each e-TAG's timer interrupt register, OCR1A, was initialized with the same value, $t$, resulting in the same calculated sampling rate. To calculate the frequency, $f$, of a signal based on $t$, the frequency of the microcontroller, $F$, and the clock prescaler, $P$, the following equation is used [24]:

$$f = (F/P)/t \tag{3.1}$$

In this application, $F = 8MHz$, $P = 64$, and $t = 1000$ for the accelerometer e-TAGs. This resulted in a target frequency of $125Hz$. For the piezoelectric e-TAGs, $t = 167$ resulting in a frequency of $748.5Hz$. In addition, to minimize traffic on the network, the piezoelectrics only sent data after 6 samples were collected, effectively reducing its frequency to $124.75Hz$.

However, due to inaccuracies within each individual clock, all e-TAGs did not have the same actual sampling rate. Over time, the clock drift associated with each e-TAG became more significant. For this application, the significance of the drift became clear when comparing the location with respect to time of a heel strike on a piezoelectric e-TAG with that of the corresponding accelerometer e-TAG, as shown in Figure 3.2.

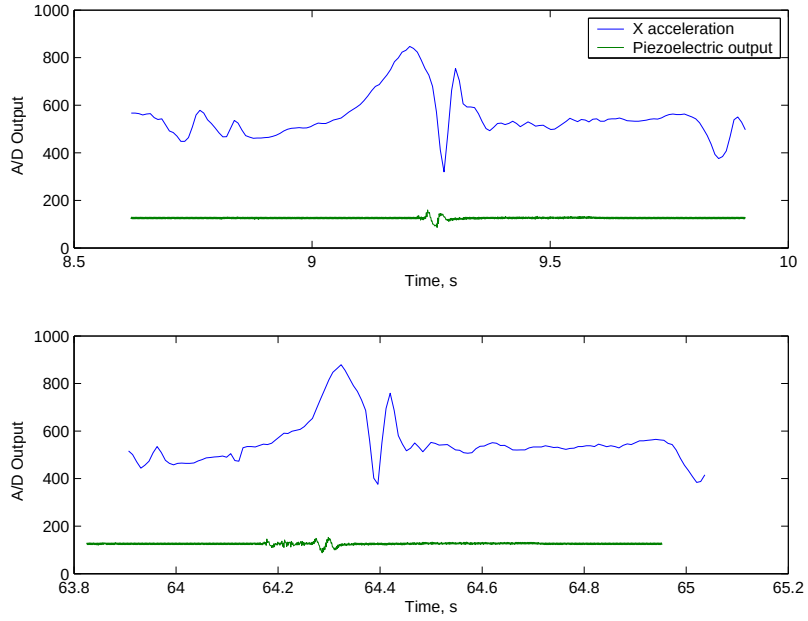


Figure 3.2: Acceleration data plotted with piezoelectric data during 2 heel strikes. The heel strike corresponds to a spike in both data sets. Notice that early on they line up and later in the same data set, they are offset from one another.

To minimize the clock drift problem, Matlab's smoothing spline, spaps, was used with a tolerance of zero [29]. An e-TAG was chosen to be the synchronization e-TAG. After a fixed number of packets, $p$, was received by the host machine from the synchronization e-TAG, all of the other e-TAGs' data over that interval was splined to be of length $p$. For this application, p was chosen to be 100. This resulted in all e-Tags having the same number of samples over the set of data, and therefore the same sampling rate. Also, by grouping them with a relatively small $p$, errors due to inaccurate clocks were minimized.

## 3.4 Accelerometer Calibration

The accelerometer chosen for this application was the ADXL203 from Analog Devices [25]. This device provided the best combination of sensitivity (1000 mV/g) and dynamic range (+/-1.7g) for the purpose of measuring step length with a two axis accelerometer. These accelerometers output an analog signal that must be converted into a digital number and further converted to an actual value in g's or $mm/s^2$. The acceleration sensed by these sensors can be divided into two different components. Gravity acts on the devices as a static component. This component is zero when an axis is oriented such that it is parallel to the ground (or perpendicular to gravity), and likewise has a value of +/-1 g when the axis is parallel to gravity. A dynamic component of the accelerometer is caused by actual movement of the sensor [30].

Each axis of each accelerometer has a slightly different sensitivity, $s$, and offset, $o$. In other words, each axis outputs a different voltage value that corresponds to 1 g, as well as a different value that corresponds to 0 g. Because of this, calibration of each accelerometer axis independently was necessary to obtain the most accurate readings from these sensors. In [31], Analog Devices described a method for calibrating a pulse width modulated accelerometer.

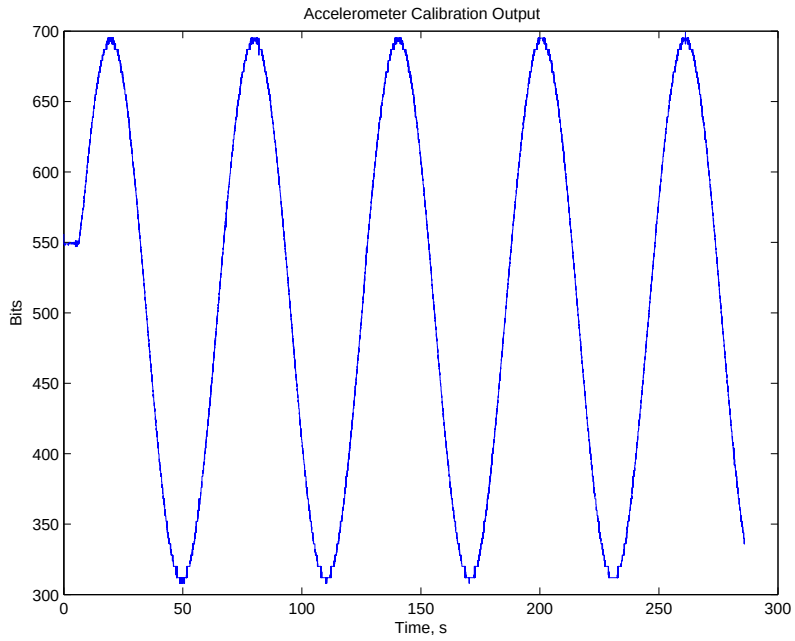


Figure 3.3: Output of an accelerometer as it is rotated 360°

This method used gravity as a frame of reference and rotated the accelerometer 360° with the sensor perpendicular to the ground, such that both positive and negative forces of gravity were applied to the sensor. It was important to rotate the device slowly, so that no rotational acceleration contributed to the reading of the sensor. When this method was applied to an ADXL203, the output resembled that of a sine wave, as shown in Figure 3.3. The output voltage of the device is defined by the following equation:

$$V = s * A + o \tag{3.2}$$

where $A$ is the acceleration in g's. The maximums of Figure 3.3 represents 1 g and the minimums represents -1 g. Using these points, it was possible to extract the values of $s$ and $o$ such that $s = (max - min)/2$ and $o = (max + min)/2$ for each accelerometer, where $max$ is the maximum voltage and $min$ is the minimum voltage.

In order to create Figure 3.3, the accelerometer had to be mounted vertically and rotated 360°. This was done with the apparatus shown in Figure 3.4. A gear was attached to the back side of the e-TAG which allowed for the sensor to be connected to a plastic rod and through a series of gears to an electric motor. A small voltage was applied to the motor to slowly but continuously rotate the accelerometer 360°.

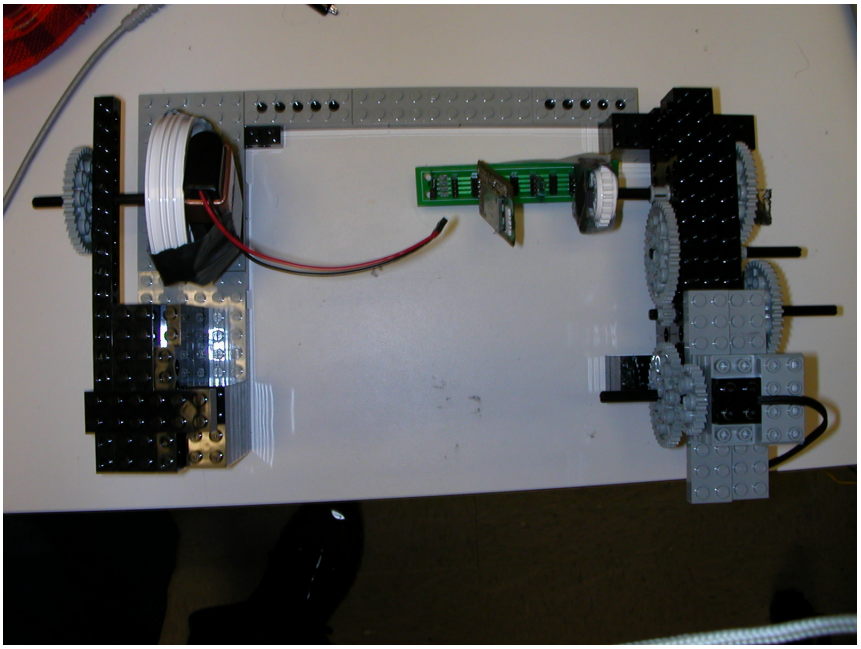

Figure 3.4: An accelerometer set up to be calibrated

## 3.5 Initial Accelerometer Corrections

For this application to produce accurate results, a few characteristics of the accelerometers must be dealt with. At this point, an explanation is necessary to understand how an accelerometer functions. It can be thought of as a spring with a weight on the end of it. If the spring is perpendicular with respect to gravity, gravity has no effect on it. However, if the spring is parallel to gravity, the full force of gravity acts on the spring, changing the

accelerometer's output, as gravity is an acceleration. The implication of this is that as the accelerometer changes orientation, the effect gravity has on each axis also changes. This had to be taken into account when processing the accelerometer's output. If the accelerometer was oriented as depicted in Figure 3.5 and $\theta$, the angle of the sensor with respect to the horizontal, was known then the following equations could be applied over each discrete sample to correct for and remove gravity from the system:

$$A_x = A_x - \sin\theta * g \tag{3.3}$$

$$A_y = A_y + \cos\theta * g \tag{3.4}$$

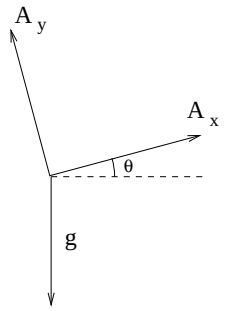


Figure 3.5: The effect of gravity on the accelerometer

Another initial issue was the angle of the accelerometer with respect to the leg. Ideally, the accelerometer would be attached so that the Y-axis was parallel to the subject's leg. However, such accuracy is quite challenging. Tilt sensing is a more common application of accelerometers such as these and is similar to finding the initial angle. By using the above calculated accelerometer parameters, $s$ and $o$, the initial angle of the accelerometer can be found. For example, an accelerometer that is perfectly aligned with the subject's leg, when the subject is standing still should output $o$ on the X-axis and $o-s$ on the Y-axis. Deviations from this can be exploited to find the initial angle. By finding the initial offset from the

values listed above, and knowing the value of $1g$ on the accelerometer, the initial angle is found. Since this is a dual axis accelerometer, two angles can be found with this method, the pitch, $\theta_x$, and the roll, $\theta_y$ [25]. The axis that is closer to perpendicular to the force of gravity is more sensitive to changes in orientation and tends to measure the tilt angle more accurately. If $i$ is the initial value output by the accelerometer when the subject is standing straight up then the initial angle, $\theta$, calculated from both axes is:

$$\theta = \sin^{-1}(\frac{i-o}{s}) \tag{3.5}$$

Figure 3.6 shows the X and Y accelerations before and after compensating for the initial the angle. Notice that initially the X acceleration was not at zero and that this is corrected in the lower plot.

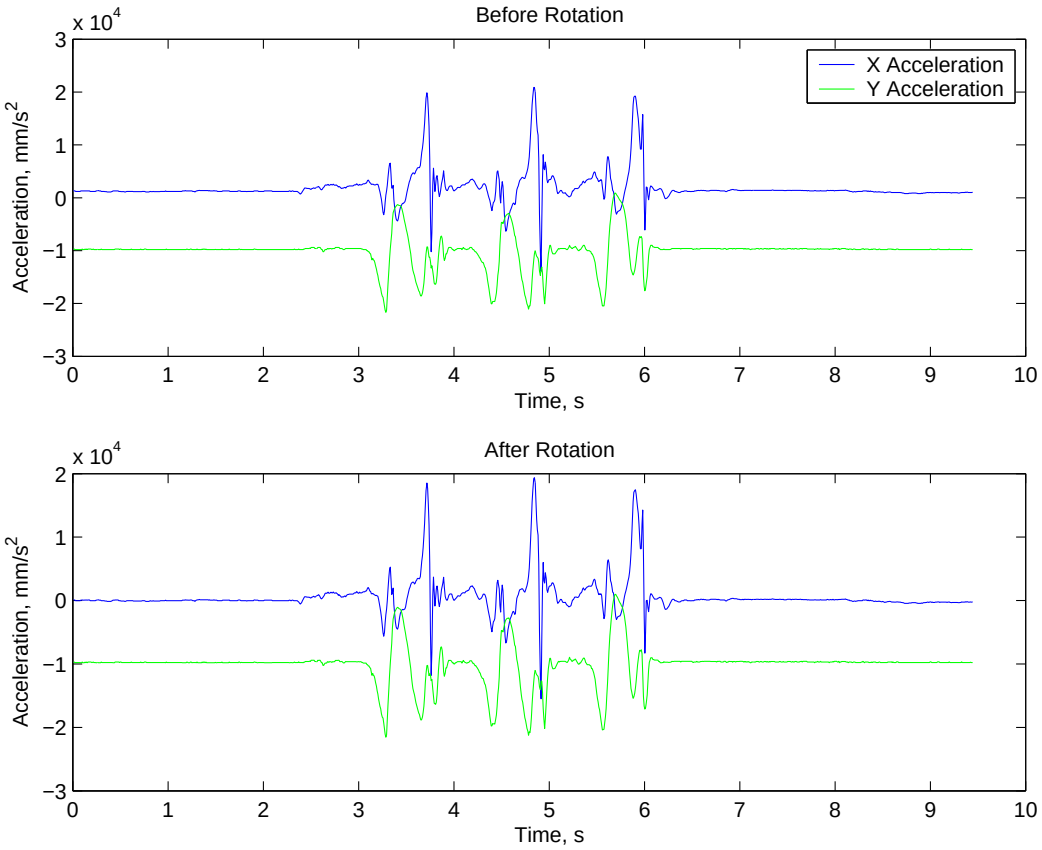


Figure 3.6: The X and Y accelerations before and after compensating for the initial angle

# Chapter 4

# Sensor Models

## 4.1 Problem Statement

A goal of this work was to be able to accurately model the gyroscope and accelerometer sensors for use in simulating future e-textile applications. Specifically, the goal was to be able to transform motion capture position data into the corresponding data that would be produced by an accelerometer and gyroscope sensor at the same position on the body. This chapter describes the models of the gyroscope and accelerometer, the motion capture data that was used to drive the simulation and the method used to validate these models.

There are many benefits to simulating e-textile applications. Simulation before prototype development allows for the same data to be applied to the model repeatedly, thus permitting the application to be refined in a deterministic manner. Also, simulation allows for design flaws to be realized early in the design process, resulting in a decrease in the time and cost of product development. Another benefit to a validated simulation is that it increases the number of data sets available for testing; this is only limited by the range of motion and

subjects available in motion capture format. Additionally, the ability to model these sensors accurately on any part of the body allows for further design space exploration before sensor placement is committed in the prototype. A validated model of these sensors would allow for years old motion capture data to be analyzed as though generated by the e-textile.

To focus the effort, sensors located on the subject's knees and ankles were modeled. However, as discussed in [4], this model can easily be expanded to any point on the body for maximum system flexibility. There, two XYZ coordinate sets, a source and a destination were used in conjunction with a translation distance $\delta d$ to interpolate a new point. This method is limited by the number of points on the body recorded during the motion capture session. Another constraint for this application is that enough points must be available to correctly estimate the angle of the location modeled, which is fundamental to both the accelerometer and gyroscope models.

## 4.2 Motion Capture Data

To stimulate the sensor models, motion capture data was used. Motion capture data records the positions in XYZ coordinates of reflective balls attached to the subject at various points throughout the body. A wide range of motion capture data is freely available in .c3d format through the Carnegie Mellon University Human Motion Capture Database project [32]. For the purpose of validating these models, motion capture data was obtained from the Virginia Tech Locomotion Research Laboratory.

In order to take derivatives of this data, as required to obtain angular velocity and acceleration from position data, it was necessary to smooth the motion capture data to remove high frequency noise from the derivatives. This noise is introduced by the error of the motion capture system, $\xi$. During discrete differentiation, the instantaneous error is increased

significantly. Discrete differentiation of a point is defined by the following:

$$V = \frac{\chi_2 - \chi_1}{\Delta t} \tag{4.1}$$

When you factor the error into this equation it becomes

$$V = \frac{(\chi_2 + \xi_2) - (\chi_1 + \xi_1)}{\Delta t} \tag{4.2}$$

Furthermore, the error associated with $V$ is

$$|\xi_3| \leq \frac{|\xi_2| + |\xi_1|}{\Delta t} \tag{4.3}$$

The error of the motion capture data was stated as being within $1mm$ and the sampling rate was $120Hz$, with a corresponding $\Delta t$ of $0.0083s$. Substituting these values into equation 4.3 results in an instantaneous error $240mm/s$. This is an 120 times increase in the magnitude of the error.

In order to correct for error, two methods of smoothing the data were considered: splining the data with Matlab's Spline toolbox [29], as discussed in [4], and using a fourth order lowpass digital Butterworth filter. One issue when splining data is the selection of the amount of tolerance to spline to. Selecting too large a tolerance will result in data loss due to excessive smoothing, while too small of a tolerance results in high frequency noise interfering with the data accuracy. The spaps spline was used for this purpose. This spline creates a B-form of a cubic smoothing spline for the given data. The smoothing tolerance was chosen such that the peaks were not excessively smoothed when compared with the actual data. The Butterworth filter was suggested as an alternative to splining the data by the Virginia Tech

Locomotion Research Laboratory. A cut off frequency of 6 Hz was used and this was the method selected to smooth the data.

## 4.3 Gyroscope Sensor Model

The gyroscope sensor model was based on the operational characteristics of Analog Devices' ADXRS300 rate sensing gyroscope [26]. The data sheet for this device states that it outputs $V_{dd}/2$ when stationary and 5 mV/(°/s) when the sensor is rotated along the axis perpendicular to its orientation and operating on a 5 V supply.

The first step in simulating this device was to find the angle of the orientation of the sensor. For the ankle application, it was assumed that the e-TAG was placed so that the Y axis of the sensor lined directly up with the subject's shin and that the axis of sensitivity for the device would be the Z axis, as shown in Figure 4.1. To calculate the angle of the sensor, two XY coordinate motion capture points were used, and a third point was interpolated to create a right triangle from these two points. The triangle was formed between the ankle's position coordinates, $(A_x, A_y)$, the knee's, $(K_x, K_y)$, and a third point, $(K_x, A_y)$, as shown in Figure 4.1. This angle, $\theta$, can be found for each discrete time sample, $t$, in the data set resulting in a function over time, $\theta(t)$. Once the function for the angle was found, the angular velocity was calculated by taking the discrete derivative of each point with respect to $dt$, the inverse of the motion capture sampling rate. A sample angle and angular velocity calculated with this method is shown in Figure 4.2.

The next step in this process was to use the information from the ADXRS300's data sheet to convert the angular velocity from $^\circ/s$ to bits or volts, which ever is desired. For this application, bits was necessary and the following equation was applied to the angular velocity, $\omega(t)$:

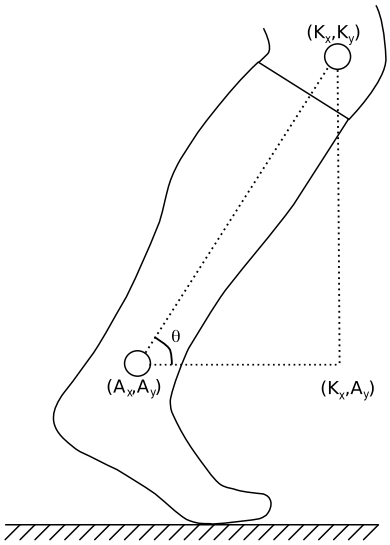


Figure 4.1: The triangle created to find angle of orientation for the gyroscope model

$$\omega(t)_{new} = \frac{\omega(t) * S * 2^R}{V_{dd}} + V_{dd}/2 \tag{4.4}$$

where $S$ is the sensitivity of the device at the operating voltage, $V_{dd}$, is as listed in the data sheet [26] and $R$ is the resolution of the analog to digital converter.

## 4.4 Accelerometer Model

The accelerometer model was based on the operational characteristics of Analog Devices' ADXL203 dual axis accelerometer [25]. A description of this device and its operating characteristics can be found in Chapter 3. In that chapter, the focus was converting raw accelerometer data to actual accelerations by accounting for the effects of gravity and orientation change. Here, the focus was to transform motion capture position data into raw acceleration data. These two operations are inverses of each other. Since this is a dual axis accelerometer, both the X and the Y accelerations were modeled, based on the orientation

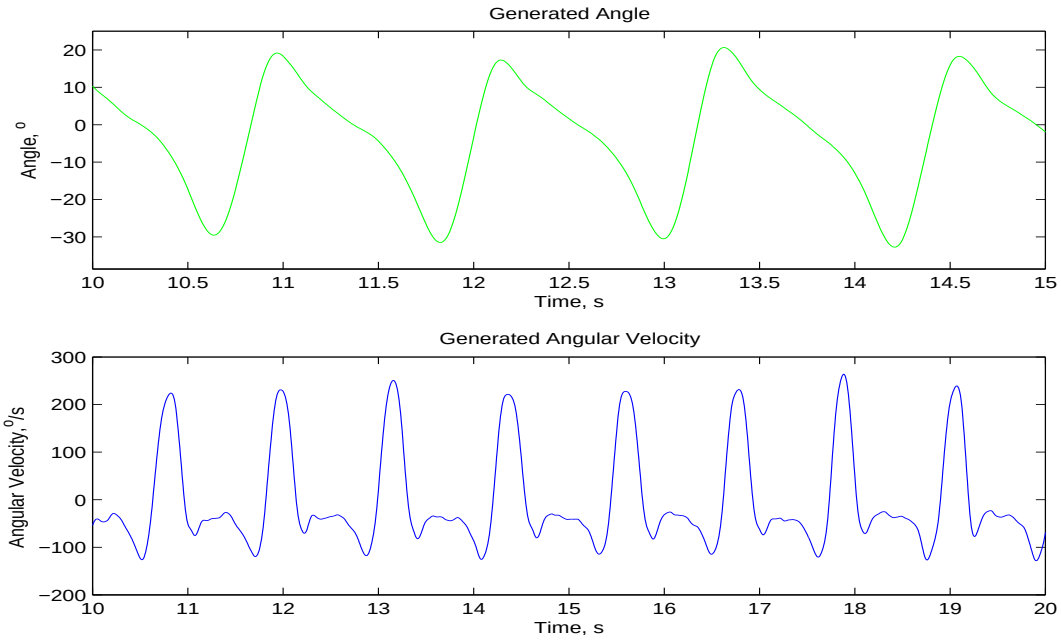


Figure 4.2: A sample angle and its corresponding angular velocity generated by the gyroscope sensor model

of the sensor on the e-textile.

Many of the techniques discussed Sections 3.5 and 5.3 were inverted and reversed in order to generate this sensor model. A raw acceleration was calculated from the position data by discretely differentiating each XY coordinate of the position data two times with respect to the time interval, $\Delta t$:

$$A_{x*} = \frac{d^2 P_x(t)}{dt^2} \tag{4.5}$$

$$A_{y*} = \frac{d^2 P_y(t)}{dt^2} \tag{4.6}$$

This raw acceleration, $A_{x*}$ and $A_{y*}$, is the acceleration in the absolute axis directions and

does not account for any change in orientation of the sensor. To account for this, the angle calculated in the above section was incorporated with these raw accelerations. Section 5.3 discusses using a rotation matrix to correct the acceleration for the change in orientation of a person's ankle during a step. This same idea was applied here to give the simulated acceleration the effects of the orientation change, but since the acceleration is being rotated the opposite way, $-\theta(t)$ was used in equations 5.3 and 5.4.

Another effect of the change in orientation of the sensor, as discussed in Section 3.5, is the effect gravity has on the sensor's output. To add this into the model, the opposite of equations 3.3 and 3.4 were applied to the accelerations:

$$A_x = A_x + \sin\theta * g \tag{4.7}$$

$$A_y = A_y - \cos\theta * g \tag{4.8}$$

At this point, the simulated accelerations match the sensor's output in everything but units and bias offset. Since the motion capture data is in $mm$, the modeled accelerations are in $mm/s^2$. Information from the ADXL203's data sheet along with each individual axes' sensitivity and offset, $s$ and $o$, was used to convert the model from these units into bits, the output of the Atmega8's analog to digital converter. This was done with the following equation for both axes:

$$A_{new}(t) = \frac{A(t) * s}{g} + o \tag{4.9}$$

where $g$ is the gravitational constant $9.81 m/s^2$.

## 4.5 Validation of Sensor Models

Once these models were developed, their functionality was verified through a direct comparison with actual accelerometer and gyroscope data. Data from various subjects was recorded in the Virginia Tech Locomotion Research Laboratory. This data was collected with the motion capture system at the same time as with the e-textile prototype pants as discussed in Section 4.5.1.

Subjects were instructed to walk on a treadmill while both mediums were collecting data. The speed of the treadmill was varied over three trials; the first speed being an individually predetermined "comfortable" speed, and the speed being increased by 10% for each of the next two trials. Ten steps of data was extracted from each data set. The subjects were classified as old or young. The five elderly subjects were between the ages of 65 and 85, and the five young subjects were between the ages of 24 and 30. The subjects consisted of four females and six males.

For each data set, the sensors on the ankles and knees were modeled as discussed in Sections 4.3 and 4.4. For simplicity, results are shown and analyzed for the right ankle in Section 4.5.2; results for the knees and left ankle are included in Appendix A.

### 4.5.1 Synchronizing Motion Capture Data with E-Textile Data

When comparing data collected through different mediums, such as via the motion capture system and with the e-textile, a method of synchronizing the data with respect to those mediums is required. To completely synchronize this data, two adjustments were necessary. First, since each medium began collecting data at slightly different times, the data needed to be aligned. Further, because of slight errors in the sampling rate, as time progressed, data

aligned early in a set became disjoint. To compensate for this, a feature detection algorithm, described below, was developed to find corresponding peaks towards the end of the data set. The e-textile data was then scaled accordingly to align completely with the motion capture data.

To detect the initial time difference, the subject was requested to perform a signal that could be easily distinguished in both data sets. In this case, the activity was the tapping of a heel three times. After the data was collected, a synchronization point was selected in each medium as shown in Figure 4.3. Both data sets were then shifted so that this point corresponded to a time value of 0.

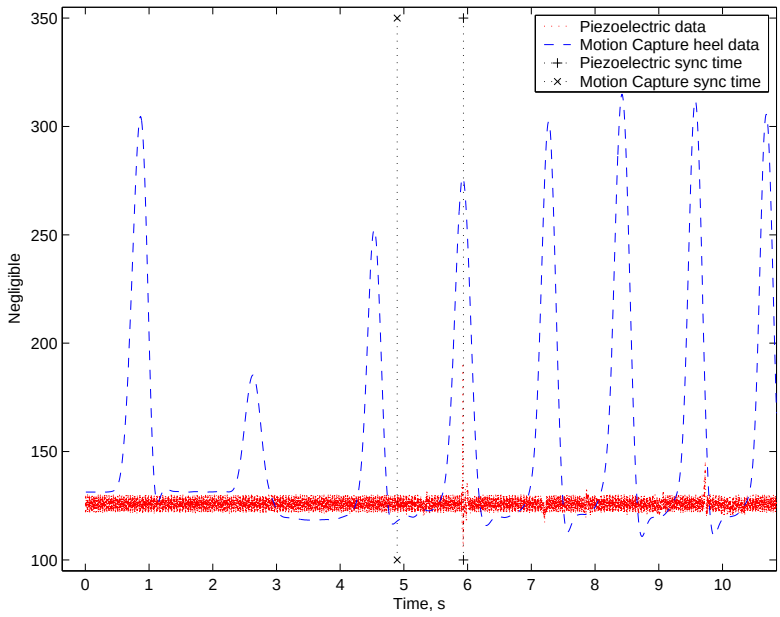


Figure 4.3: Piezoelectric heel strike data overlaid with motion capture heel position data. The vertical lines show the selected synchronization points for each. A heel strike is a spike in the piezoelectric data, and is a local minima in the motion capture heel data.

During walking, the data from the accelerometers and the gyroscopes follow a regular pattern. This was exploited in the feature detection algorithm to adjust the frequency of the data to completely align both mediums. The gyroscope data has a regular peak with each step that

corresponds to the sharp increase in angular velocity during the forward motion of a step. The final peak of the gyroscope in the medium with the shortest data set was found, and using that time, a corresponding peak was found in the other medium's data set, as shown in Figure 4.4. The following equation was used to calculate a new frequency for the e-textile data:

$$F_{new} = F * (1 - \frac{p_m - p_e}{p_m}) \tag{4.10}$$

where $p_m$ and $p_e$ are the peak times for the motion capture and e-textile data respectively, $F$ is the original e-textile frequency, and $F_{new}$ is the new e-textile frequency.

Figure 4.5 shows each the results of this process.

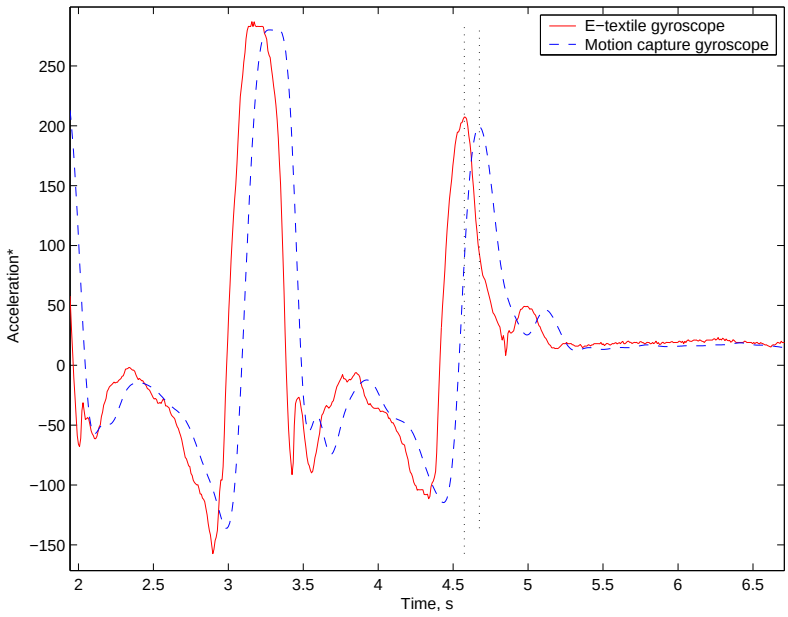


Figure 4.4: Common peaks identified in the e-textile and motion capture gyroscope data

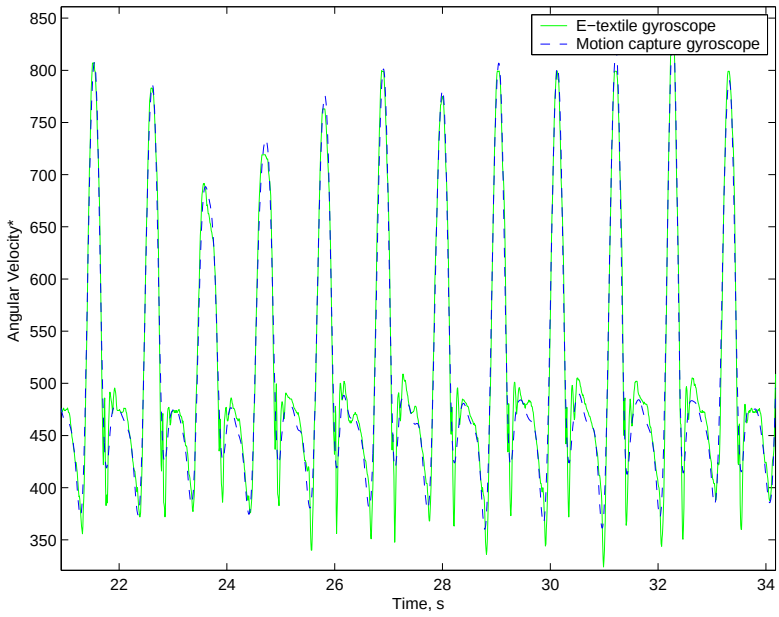


Figure 4.5: Synchronized motion capture and e-textile gyroscope data.

### 4.5.2 Results

A major focus of this thesis was simulating two commonly used motion sensors, an angular rate sensing gyroscope and a dual axis accelerometer. Figures 4.6-4.8 show the modeled angular velocities and accelerations compared visually with the raw output of each of the respective sensors over five steps for the right ankle.

The correlation coefficients, $r$, between the motion capture data and the raw data are shown in Table 4.1.[1] The null hypothesis is that the motion capture and the raw data are not related, and as can be seen by the high significance of the correlation and the low probability, the null hypothesis was proven false. It is important to note that correlation only "measures the strength of the relation between the two variables, not the agreement between them" [33]. Furthermore, one would expect that two methods of measuring the same quantity would be

[1]In all tables included in this thesis, $A_x$ is the X axis acceleration, $A_y$ is the Y axis acceleration, and $\omega$ is the angular velocity.

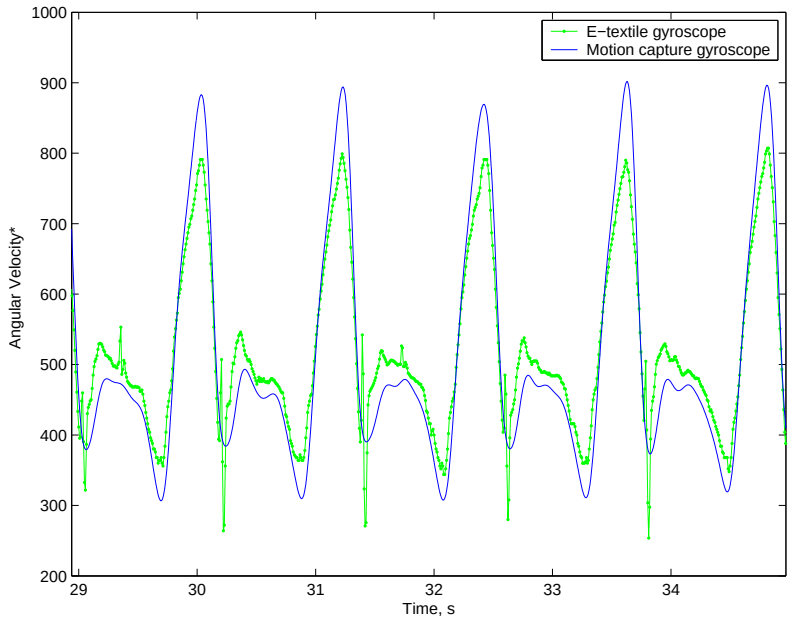


Figure 4.6: The motion capture angular velocity overlaid with the raw gyroscope output

strongly related. With that said, correlation alone is insufficient to prove that the motion capture data agrees with the raw output.

Table 4.1: Correlation between e-textile and motion capture data for right ankle data

| | Pearson Correlation | 2-Tailed Significance |
|---|---|---|
| $A_x$ | 0.834 | 0 |
| $A_y$ | 0.757 | 0 |
| $\omega$ | 0.970 | 0 |

In [33], Bland presents a method for assessing agreement between two methods of measurement. This method is based on calculating "limits of agreement" between the two methods. The limits of agreement are defined by:

$$[u, l] = \bar{d} \pm 1.96s \tag{4.11}$$

where $u$ and $l$ are the upper and lower limits of agreement, $\bar{d}$ is the mean of the differences

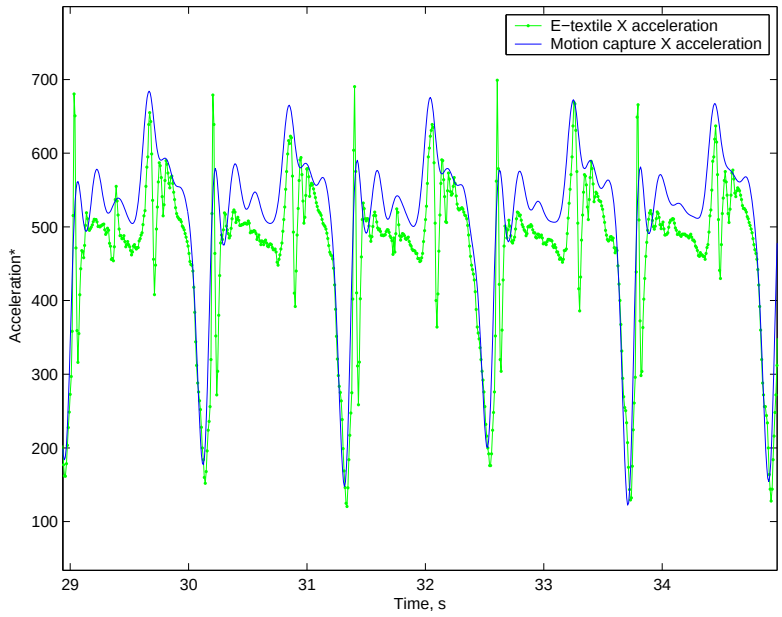


Figure 4.7: The motion capture X acceleration overlaid with the raw X acceleration output

between the two methods, and $s$ is the standard deviation of those differences. $\bar{d}$ is known as the bias between the two methods. Since on average, this is the difference between the two methods, this bias can then be accounted for in either of the methods as necessary.

The limits of agreement for the right ankle data are shown in Table 4.2. Ideally, if the two methods were in complete agreement, both $\bar{d}$ and $s$ would be zero, resulting in $[u, l] = [0, 0]$. Also included in this table is the mean of the differences, $\bar{d}$, its absolute value, the standard deviation and standard error of both means. From this data it is possible to derive both a real and a relative error between the raw data and the motion capture data. This is the error after removing the bias. If $s$ is the standard deviation, $s_e$ is the standard error of $s$, then the real error, $\epsilon$, is $\pm(2s + s_e)$. Furthermore, the relative error with respect to the motion capture data is $\epsilon/R$ where $R$ is the range of the motion capture data, shown in Table 4.2. Each of these measures can be found in Table 4.3. Since the limits of agreement include this bias, the relative error actually gives more information about how closely these two methods

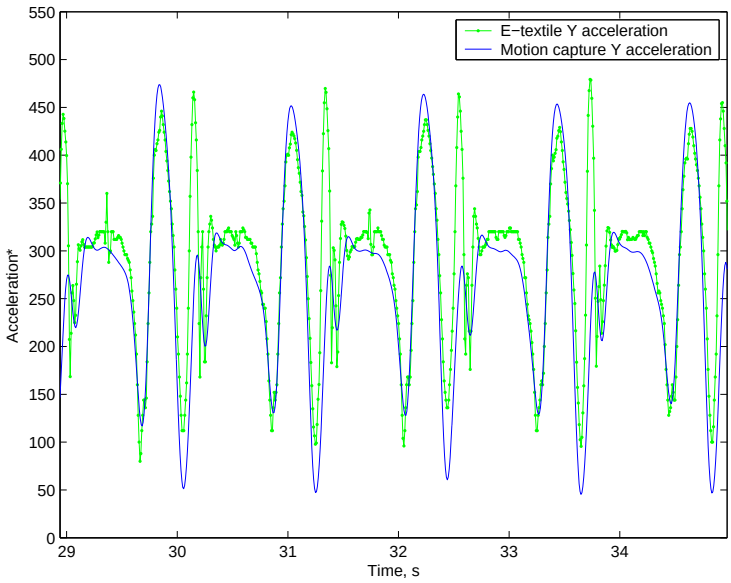


Figure 4.8: The motion capture Y acceleration overlaid with the raw Y acceleration output

agree.

Table 4.2: The limits of agreement and its confidence interval for the right ankle data

| | $d$ | | | $\|d\|$ | | |
|---|---|---|---|---|---|---|
| | avg | std | error | avg | std | error |
| $A_x$ | -63.383 | 73.462 | 0.387 | 75.849 | 60.507 | 0.319 |
| $A_y$ | 23.415 | 70.068 | 0.369 | 47.252 | 56.790 | 0.299 |
| $\omega$ | 3.882 | 41.578 | 0.219 | 30.363 | 28.668 | 0.151 |
| | Limits of Agreement | | 95% confidence interval | | | motion capture |
| | lower | upper | lower | upper | n | range |
| $A_x$ | -207.369 | 80.603 | -208.683 | 81.916 | 36050 | 890.028 |
| $A_y$ | -113.919 | 160.749 | -115.171 | 162.001 | 36050 | 704.859 |
| $\omega$ | -77.611 | 85.375 | -78.354 | 86.118 | 36050 | 727.704 |

Included in these results are tests of within-subjects and between-subjects effects. Between-subjects effects are the effects of independent variables which vary from subject to subject but remain the same for all observations of a subject. Within-subject variables are those that can vary from observation to observation for a particular subject. For this application,

Table 4.3: The real and relative error of the two methods after removing the bias, with respect to the range of the motion capture data for the right ankle data.

| | std error | real error | relative error |
|---|---|---|---|
| $A_x$ | 0.670 | $\pm$147.595 | $\pm$0.166 |
| $A_y$ | 0.639 | $\pm$140.776 | $\pm$0.200 |
| $\omega$ | 0.379 | $\pm$83.535 | $\pm$0.115 |

the within-subjects variable is the speed of the trial. The between-subjects effects are the age and sex of the subject. Tables 4.4-4.6 show the between-subjects effects. The critical statistic in these tables is the mean significance. In order for any of these effects to be significant at the $\alpha = 0.05$ level, the significance must be below that. Therefore, it can be concluded that neither of the between-subjects effects are significant for either acceleration or angular velocity. Tables 4.7-4.9 show the tests of within-subjects effects. A multivariate analysis of variance, Pillai's trace, was used to generate these statistics. More information on this can be found in [34] and [35]. Since speed alone is not significant in any of the tables, the other tests, such as speed*age can be ignored. They are included here for clarity and completeness. Based on the high significance of each of these effects, it can safely be concluded that there was no significant effect on the agreement of these two methods caused by any of the independent variables.

### 4.5.3 Analysis

The previous section presented the results of using motion capture data to simulate an ADXL203 dual axis accelerometer and an ADXRS300 rate sensing gyroscope. Table 4.3 showed that, after removing the bias between the two measures, the angular velocities were within 12% of each other and the accelerations were within 16% and 20% respectively for $A_x$ and $A_y$ of the right ankle. As shown in Appendix A, these results were consistent across

Table 4.4: Tests of between subject effects of independent variables, right ankle $\omega$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0 | 1 | 0 | 0.339 | 0.579 |
| | maximum | 0.012 | 1 | 0.012 | 0.598 | 0.465 |
| | std dev | 0 | 1 | 0 | 0.413 | 0.541 |
| sex | mean | 2.93e-05 | 1 | 2.93e-05 | 0.025 | 0.879 |
| | maximum | 0.003 | 1 | 0.003 | 0.137 | 0.722 |
| | std dev | 7.77e-05 | 1 | 7.77e-05 | 0.177 | 0.687 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.008 | 7 | 0.001 | | |
| | maximum | 0.136 | 7 | 0.019 | | |
| | std dev | 0.003 | 7 | 0 | | |

Table 4.5: Tests of between subject effects of independent variables, right ankle $A_x$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.008 | 1 | 0.008 | 1.859 | 0.215 |
| | maximum | 0.023 | 1 | 0.023 | 3.049 | 0.124 |
| | std dev | 0.002 | 1 | 0.002 | 2.085 | 0.192 |
| sex | mean | 0.003 | 1 | 0.003 | 0.772 | 0.409 |
| | maximum | 0.002 | 1 | 0.002 | 0.228 | 0.647 |
| | std dev | 0.001 | 1 | 0.001 | 0.486 | 0.508 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.029 | 7 | 0.004 | | |
| | maximum | 0.052 | 7 | 0.007 | | |
| | std dev | 0.007 | 7 | 0.001 | | |

Table 4.6: Tests of between subject effects of independent variables, right ankle $A_y$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0 | 1 | 0 | 0.047 | 0.834 |
| | maximum | 0.207 | 1 | 0.207 | 3.741 | 0.094 |
| | std dev | 0.007 | 1 | 0.007 | 2.652 | 0.147 |
| sex | mean | 0.001 | 1 | 0.001 | 0.261 | 0.625 |
| | maximum | 0.057 | 1 | 0.057 | 1.027 | 0.345 |
| | std dev | 0.002 | 1 | 0.002 | 0.699 | 0.431 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.019 | 7 | 0.003 | | |
| | maximum | 0.387 | 7 | 0.055 | | |
| | std dev | 0.018 | 7 | 0.003 | | |

Table 4.7: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right ankle $\omega$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.626 | 1.976 | 6 | 26 | 0.106 |
| speed*age | 0.758 | 2.643 | 6 | 26 | 0.039 |
| speed*sex | 0.419 | 1.147 | 6 | 26 | 0.364 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table 4.8: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right ankle $A_x$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.418 | 1.146 | 6 | 26 | 0.364 |
| speed*age | 0.53 | 1.562 | 6 | 26 | 0.198 |
| speed*sex | 0.506 | 1.468 | 6 | 26 | 0.228 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table 4.9: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right ankle $A_y$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.456 | 1.278 | 6 | 26 | 0.302 |
| speed*age | 0.366 | 0.971 | 6 | 26 | 0.464 |
| speed*sex | 0.334 | 0.869 | 6 | 26 | 0.53 |
| speed*age*sex | 0 | . | 0 | 0 | . |

the left ankle as well. The relative error of $A_y$ for the left ankle was slightly higher at 25% due to a higher standard deviation of the difference between the two data sets.

There were a few expected sources of error in the simulation of these sensors. As discussed in Section 3.5, because of gravity, the initial angle of the accelerometer significantly effects its output. When simulating the accelerations, it was assumed that the accelerometer's initial angle was zero degrees. There were cases when the initial angle was obviously not zero; an example is shown in Figure 4.9. Based on typical values for $s$ and $o$ (see Section 3.5), the bias for $A_x$ should be near 512, and the bias for $A_y$ should be approximately 307. Any significant deviation from this means the initial angle assumption is inaccurate. Since there was no time in the data set when the subject stood sufficiently still, the appropriate angle adjustment could not be made. A method for calibrating and realizing the initial angle of a three-axis accelerometer is described in [30].

Another significant source of error stemmed from the specified measurement range of both the accelerometers and the gyroscopes. The accelerometer's stated range was $\pm 1.7g$, and it output $1V/g$ with a null bias of $2.5V$ when operating at $5V$ [25]. Therefore, the bounds in volts were $2.5V \pm 1.7V$. To convert this to digitized volts, the ratio $1024/5V$ was used.[2] This resulted in a range of 696.32 for the accelerations. A similar analysis was completed for the ADXRS300 rate sensing gyroscope. Its stated range was $\pm 300°/s$, with a null bias

[2]The Atmega8 had a 10-bit A/D converter, therefore the maximum voltage corresponded to $2^{10}$ or 1024. In all figures included in this thesis, a '*' denotes the output is directly from the Atmega8's A/D converter.

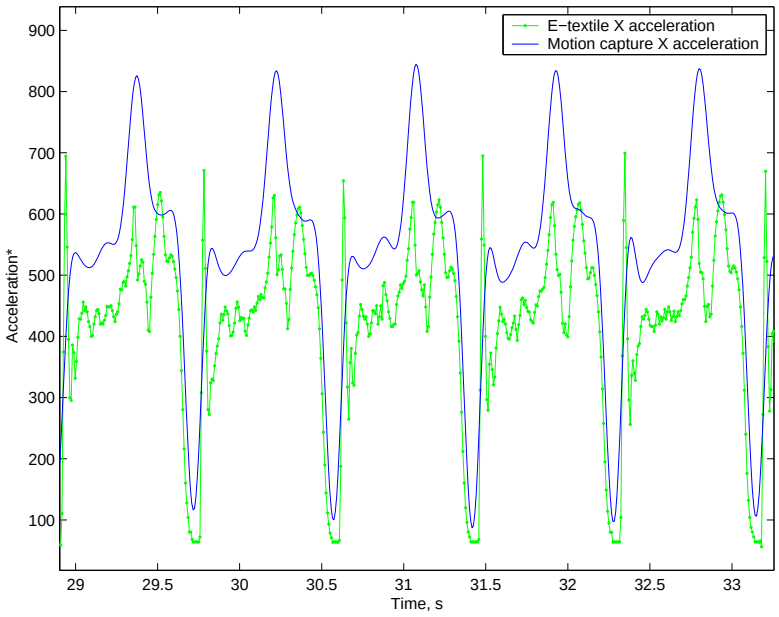


Figure 4.9: An example of the initial accelerometer angle being not equal to zero. Notice how the acceleration is consistently lower then the simulated version.

of $2.5V$ and output $0.005V/(^\circ/s)$ [26]. Therefore its bounds in volts was $2.5V \pm 1.5V$ and when digitized, the range became 614.4. According to each sensors' respective data sheets, behavior outside of these ranges was indeterminate and usually resulted in clipping of the signal. Table 4.2 shows the range of values for the simulated motion capture data. These ranges show angular velocities and accelerations significantly outside of the range of these sensors. Table 4.10 shows the percent of motion capture data points for each sensor that clipping potentially would have affected. For the ankles, notice that the Y acceleration would clip the most frequently. This was because the axis that senses the Y acceleration already had the full force of gravity acting upon it. This means that $1g$ of the $1.7g$ range of the sensor is already used, and it can only sense an additional $0.7g$.

In addition to the range issues discussed above, each sensor had a certain amount of error associated with it. Both the gyroscope and the accelerometer were slightly effected by the temperature of the sensor, as stated in [25] and [26]. Also, both sensors were designed to

Table 4.10: Percent of motion capture data points that would be clipped had clipping been incorporated into the model.

| Sensor | # of data points clipped | % of data points clipped |
|---|---|---|
| Right ankle $A_x$ | 1150 | 3.19 |
| Right ankle $A_y$ | 5492 | 15.23 |
| Right ankle $\omega$ | 3469 | 9.63 |
| Left ankle $A_x$ | 809 | 2.24 |
| Left ankle $A_y$ | 5320 | 14.73 |
| Left ankle $\omega$ | 3614 | 10.00 |
| Right knee $A_x$ | 887 | 2.46 |
| Right knee $A_y$ | 292 | 0.81 |
| Right knee $\omega$ | 3469 | 9.62 |
| Left knee $A_x$ | 852 | 2.36 |
| Left knee $A_y$ | 246 | 0.68 |
| Left knee $\omega$ | 3614 | 10.00 |

operate at $5V$ and that is the optimal operational voltage. The accelerometer operated at $3.3V$ and the gyroscope operated at $5V$ but the signal was then scaled down to $3.3V$ by a voltage divider for use with the Atmega8, which operated at $3.3V$.

Each sensor had a certain amount of noise associated with its output, described by a noise density in the data sheet. The root mean square (RMS) of the noise was found with:

$$noise_{rms} = N_d * \sqrt{BW * 1.6} \tag{4.12}$$

where $BW$ was the bandwidth of the sensor and $N_d$ is the noise density in $\mu g/\sqrt{Hz}$ for the gyroscope and $(^\circ/s)/\sqrt{Hz}$ for the accelerometer [25]. For the accelerometer, a $5Hz$ bandwidth was chosen and the noise density from the data sheet was $110\mu g/\sqrt{Hz}$. This resulted in a RMS noise value of $0.311mg$ or $3.04mm/s^2$. For the gyroscope, the bandwidth was $40Hz$ and the noise density was listed as $0.1(^\circ/s)/\sqrt{Hz}$, resulting in a typical noise value of $0.8^\circ/s$. To find the best estimate of the uncertainty of a single measurement, the

peak-to-peak noise values are typically used. This noise follows a Gaussian distribution for the accelerometer [25] and the same was assumed for the gyroscope. These values represent the standard deviation of the noise of each sensor and the mean of the noise is zero. To find the peak-to-peak noise, a 99.9% confidence interval was calculated by multiplying the RMS noise values by 6.18 as described in [36]. The peak-to-peak noise value for the accelerometer was $18.788 mm/s^2$ and $4.954°/s$ for the gyroscope. To convert these values to those that would result from the analog to digital conversion, equations 4.9 and 4.4 were applied. This resulted in 0.392 for the accelerometer and 5.073 for the gyroscope.

A final source of error involves the shock caused by the foot when the heel strikes the ground. This shock is detected by each sensor but not the motion capture data. These can be seen in Figures 4.6-4.8. Each figure has five spikes in the e-textile data at regular intervals, one for each step, that is not present in the motion capture data. There were also smaller, but significant spikes that occur during the heel lift portion of the step, as discussed in Section 5.4.

The results for the knees were significantly worse then that of the ankles. The main reason for this was that the approximation used to generate the knee's angle of orientation was fundamentally flawed. The same angle used for the ankle was used for the knee, under the theory that the sensor would be just below the knee, along the same line as shown in Figure 4.1. This approximation turned out to be quite accurate for the ankle, but not for the knee. This inaccuracy is shown via the angular velocity in Figure 4.10. Since both models fundamentally depend on an accurate angle approximation, this error propagated through to the accelerations, resulting in large relative errors.

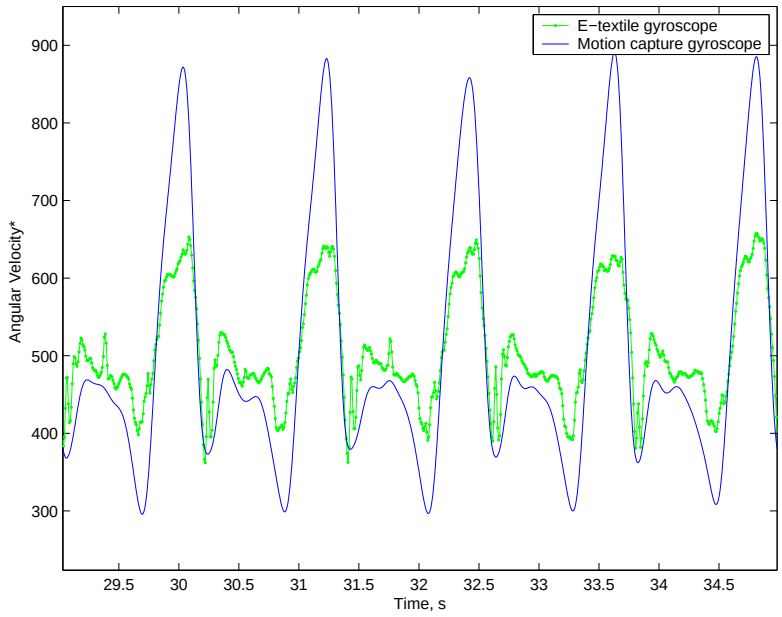


Figure 4.10: An example showing the inaccuracy of the knee angle approximation.

# Chapter 5

# Step Length Calculation

## 5.1 Problem Statement

One goal of this work is to improve upon the step length results in [4]. As described in Section 2.2, Edmison proposed placing an accelerometer on the subject's ankle and using the X-axis acceleration along with a three point, piece-wise linear fit to calculate the step length. The linear fit was necessary due to a discrepancy in the acceleration caused by the change of orientation in the accelerometer.

Though this linear fit improved the results, there were some shortcomings. First of all, certain subjects had large, unexplained errors in the step length measurement. This was usually due to the subject walking abnormally, such as a subject who shuffles. A person who shuffles does not change the orientation of the accelerometer as much as someone with a more standard gait. The correction algorithm was not as successful in this situation. Because of these shortcomings, using this correction algorithm is undesirable. Instead, it would be better to measure the orientation change and combine this with the accelerometer output.

Since the change in orientation was faulted for the measurement error in [4], placing a gyroscope on the ankle as well as the accelerometer is proposed here. A gyroscope outputs the angular velocity of the sensor, and integrating once results in the angle of the sensor. This can be used in combination with both the X and Y axes of the accelerometer to correct for the orientation of the accelerometer.

## 5.2 Design Variables

There were multiple issues that arose with adding a gyroscope to account for the accelerometer's orientation. The first was that of integration constants. In the previous attempt, only one accelerometer axis was integrated twice. Since the magnitude of step length is the target measure, the integration constant for the position measurement is zero. That left only one unknown integration constant, the initial velocity. It was assumed also that the integration constant for the velocity was also zero, since at the beginning and end of a step the heel velocity is zero. This seems like a valid assumption, however, since the sensor is actually slightly above the ankle, there is still movement of the lower shin during all facets of a step.

In this second attempt at step length, instead of integrating only one axis of the accelerometer, both the X and Y axes were integrated in order to combine them into one real world X-axis acceleration. This doubled the number of unknown integration constants: $V_{x0}$, $V_{y0}$, $P_{x0}$, and $P_{y0}$, corresponding to the initial X velocity, the initial Y velocity, the initial X position, and the initial Y position. Since we were measuring step length, the initial position is (0,0) and the latter two values listed were zero. Another integration constant that was unknown was the initial angle of the sensor, $\theta_0$, as shown in Figure 5.1. In that figure, the x-y axis was the orientation of the accelerometer, x*-y* was the real world coordinate system and $\theta_0$ was used to translate between them. This angle coincided with the orientation of the

subject's leg with respect to gravity, that is, if a subject's leg was vertical, the angle would be 0 and if the subject were to stick his leg straight out in front of him, the angle would be 90°.

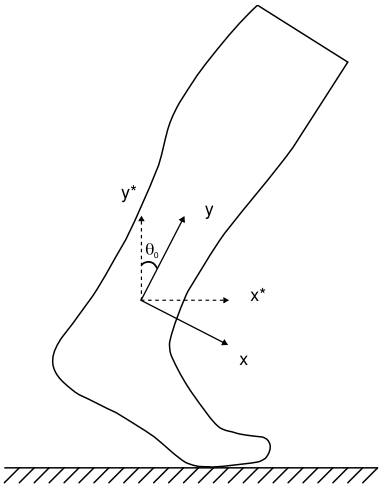


Figure 5.1: The angle of a person's foot at the beginning of a step

In order to get accurate step length from acceleration, an important decision involved choosing the integration boundaries. There are a few options in this regard, as shown in Figure 5.2. The piezoelectric strips attached to the subject's heel allowed for the time of the heel strike to be automatically found. The simplest option was to integrate from heel strike to heel strike. In the previous work [4], however, the heel strike was not included in the integration because when the heel strikes, a spike in the acceleration was caused by the shock and this could negatively effect the accuracy if included in the integration. This spike in the acceleration can be seen in Figure 5.2 directly following the "heel strike to heel strike" boundary to condition. Another idea, motivated by the desire to eliminate the initial angle integration constant, $\theta_0$, was to find the point where the leg is vertical. Visual inspection of a step shows that at some point while the foot is on the ground shortly after the heel strikes, the leg is vertical and $\theta_0 = 0$. When the leg is vertical and the heel on the ground, the acceleration of that leg should be close to zero. If the acceleration is zero and the leg is vertical, then

$A_x$ should be equal to zero and $A_y$ should have gravity acting on it completely (or be zero if gravity was already accounted for). Using this knowledge, a search was conducted to find the "vertical leg point", that is, the point where $A_x = 0$ and $A_y = -g$. Other variations of boundary conditions are also possible, such as those that do not include the heel strike. One option is to integrate from this "vertical leg point" to right before the heel strike. At this point, for simplicity's sake, the integration bounds will be from heel strike to heel strike, ignoring any effect the shock of the heel strike has on the integration.

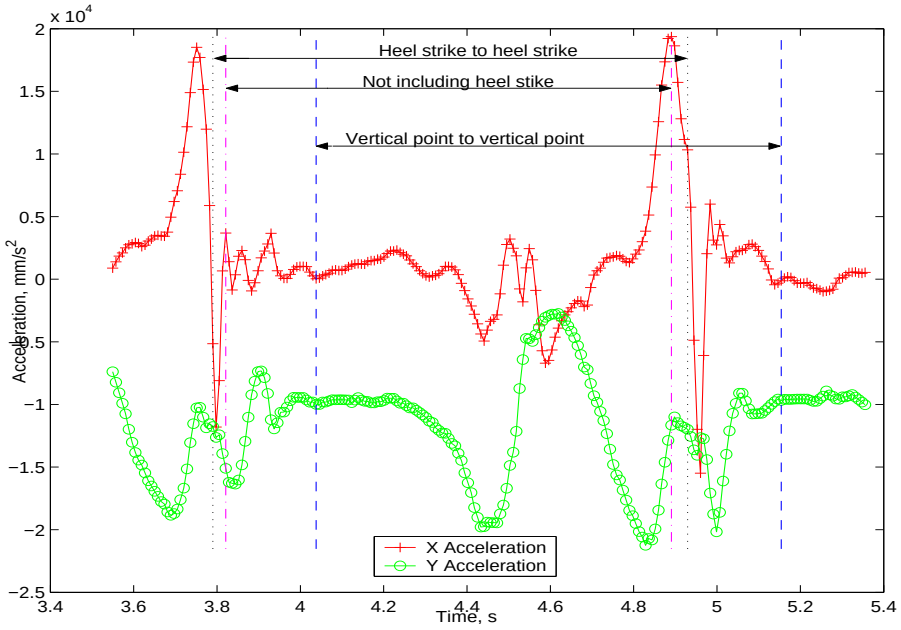


Figure 5.2: The different integration boundary possibilities when calculating step length

## 5.3 Algorithm Design

This section describes the approach taken to fuse the multiple sensors' data together to calculate a subject's step length.

If it is assumed that all the velocity and position integration constants are zero, that leaves

only one integration constant unknown, $\theta_0$; this was the initial approach taken. To find the initial angle, an iterative search was used. A best initial angle was chosen from this. In this case, a "best" initial angle was defined as one that caused the X velocity, $V_x$, to return to zero and also the Y position, $P_y$ to return to zero. That is, $\theta_0$ was chosen such that it minimizes $\eta$ where:

$$\eta = |V_x| + |P_y| \tag{5.1}$$

An initial algorithm for calculating the step length is as follows:

1. Correct for the initial angle of the accelerometer on the leg
2. Find integration interval
3. For each initial angle, $\theta_{test}$,
    a. Integrate angular velocity to get the angle of the leg over time
    b. Remove gravity from accelerations
    c. Rotate the X and Y accelerations through a rotation matrix to correct for orientation changes
    d. Integrate to get positions and velocities at this angle
    e. Check to see if $\theta_{test}$ is the "best" angle, $\theta_{best}$, as defined above
4. Calculate step length using $\theta_{best}$

See Sections 3.5 and 5.2 for discussion of items 1, 2 and 3b.

As a person walks, the angle depicted in Figure 5.1 is constantly changing and accordingly the orientation of the accelerometer is changing. In order to calculate the step length, the

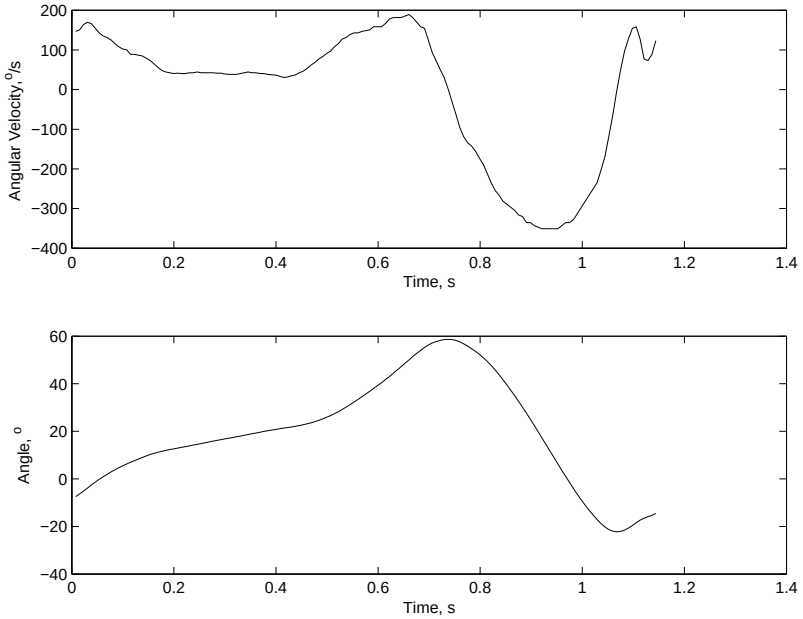


Figure 5.3: The angular velocity of a subject's ankle during a step and its corresponding angle over time

acceleration in the absolute X direction, $A_{x*}$, must be known, as opposed to the acceleration in the direction that the sensor is oriented, $A_x$. To reorient $A_x$ into $A_{x*}$, the angle of the subject's leg over time must be known. Using $\theta_{test}$ as the initial condition, the angular velocity, $\omega(t)$, was integrated to calculate the angle, as shown in Figure 5.3. So, the angle is given by:

$$\theta(t) = \int_{t_{start}}^{t_{end}} \omega(t)dt + \theta_{test} \tag{5.2}$$

After $\theta(t)$, the angle over time, was found, the next step was to use that angle along with the accelerometer's output to find $A_{x*}$ and $A_{y*}$ over the time interval. Figure 5.4 shows the the accelerations, the angle and their relationships with the absolute X and Y directions. From the figure, it can be shown that:

$$A_{x*}(t) = A_y(t)\sin\theta(t) + A_x(t)\cos\theta(t) \tag{5.3}$$

and

$$A_{y*}(t) = A_y(t)\cos\theta(t) - A_x(t)\sin\theta(t) \tag{5.4}$$

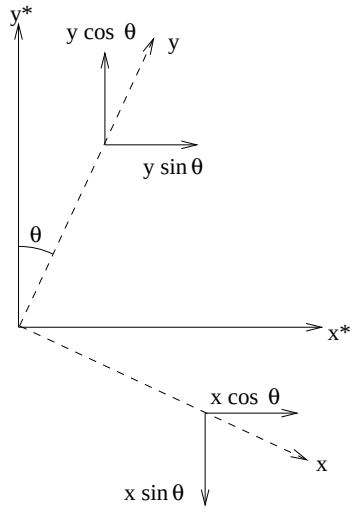


Figure 5.4: Depiction of components used for rotation matrix

After finding the accelerations $A_{x*}$ and $A_{y*}$, the velocity and position of the subject over time can be found with:

$$V_x(t) = \int_{t_{start}}^{t_{end}} A_{x*}(t)dt \tag{5.5}$$

$$V_y(t) = \int_{t_{start}}^{t_{end}} A_{y*}(t)dt \tag{5.6}$$

$$P_x(t) = \int_{t_{start}}^{t_{end}} V_x(t)dt \tag{5.7}$$

$$P_y(t) = \int_{t_{start}}^{t_{end}} V_y(t)dt \tag{5.8}$$

The subject's step length would then be $S = P_x(t_{end})$. $P_y(t)$ was used to iteratively search for the best initial angle as described above.

## 5.4 Step Length Results

Figures 5.6 and 5.7 show typical accelerations and positions that resulted from the step length algorithm discussed in Section 5.3 and Figure 5.5 shows the accelerations after gravity has been removed but before accounting for the change in orientation of the sensor. This approach did not produce results as accurately as expected. There are a few possible reasons for this. First of all, the assumption that the initial velocity in both the X and Y direction is zero at the beginning of a step is questionable. The heel's velocity should be zero at that time, but not necessarily the shin directly above the ankle, where the sensor is actually located. It is possible that there was some velocity of the accelerometer at this time. There could also be error associated with the initial angle used when the angular velocity was integrated. In addition, each of these sensors has a certain amount of error associated with it. When merging the three sensor outputs together, the error compounds and hurts the overall accuracy of this application. Also, due to the summing characteristic of discrete integration, small deviations close to zero at the beginning of the integration interval can also effect the final result. The remainder of this section is spent exploring and discussing these issues by incorporating the simulation environment discussed in Chapter 4.

Applying the sensor models to this application allowed for the step length algorithm to be simplified by removing a few of the variables discussed in Section 5.2. Specifically, the motion

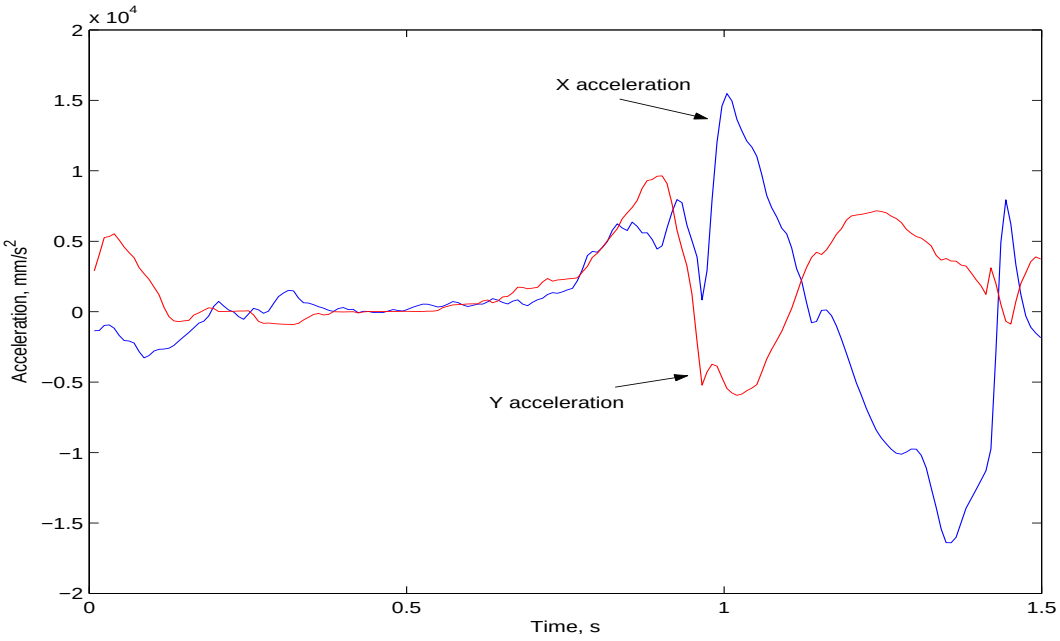


Figure 5.5: The resulting accelerations after removing gravity but before accounting for the angle of the sensor.

capture data was used along with the sensor models to eliminate both the initial velocity and the initial angle integration constants. This provided a way to verify the algorithm, as well as gain insight on the effect of the summing characteristic of discrete integration. Figures 5.8-5.11 show the resulting angles, accelerations, velocities and positions of a simulated accelerometer/gyroscope e-TAG. For comparison purposes, each graph included in this section from this point forward has the corresponding motion capture data plotted with it. For example, the motion capture acceleration was calculated by taking the second derivative of the position data. The simulated accelerations were generated from the sensor models and input into the step length algorithm. Notice how the simulated data corresponds exactly to the motion capture data. Since the same motion capture data was used to generate the sensor models, this data should and does line up exactly. This example serves as a validation of the step length algorithm when all integration constants are known because Figure 5.11

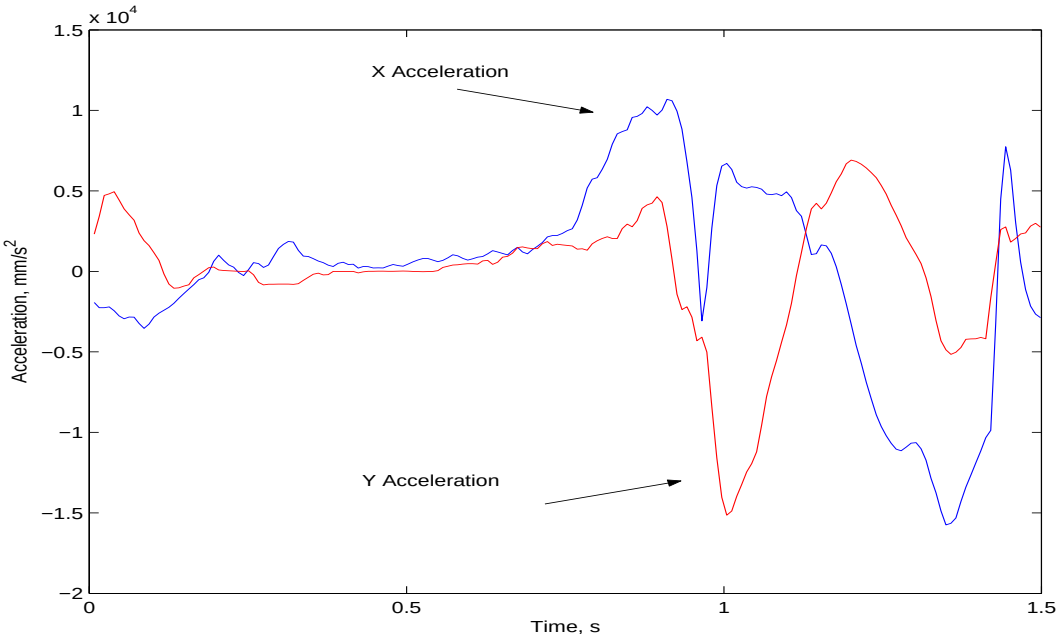


Figure 5.6: The resulting accelerations after using the algorithm in discussed in Section 5.3

shows the correct positions for both the X and Y directions.

The next analysis is similar to the previous one, but instead of using simulated sensor data, actual data from the e-textile was used. The motion capture data was still incorporated to eliminate the unknown integration constants. Figures 5.12-5.17 show these results. This example illustrates how a slight error at the beginning of the integration negatively effects algorithm. Notice how the X acceleration in Figure 5.13 was slightly below zero for the first $0.75s$ of the step. When this gets integrated into velocity, an even larger dip below zero occurs in Figure 5.15. Finally, in Figure 5.17 this error propagates through to the X position resulting in a negative step length and a corresponding step length error of 124%! This error was calculated as:

$$\frac{S_{mc} - S_{et}}{S_{mc}} * 100 \tag{5.9}$$

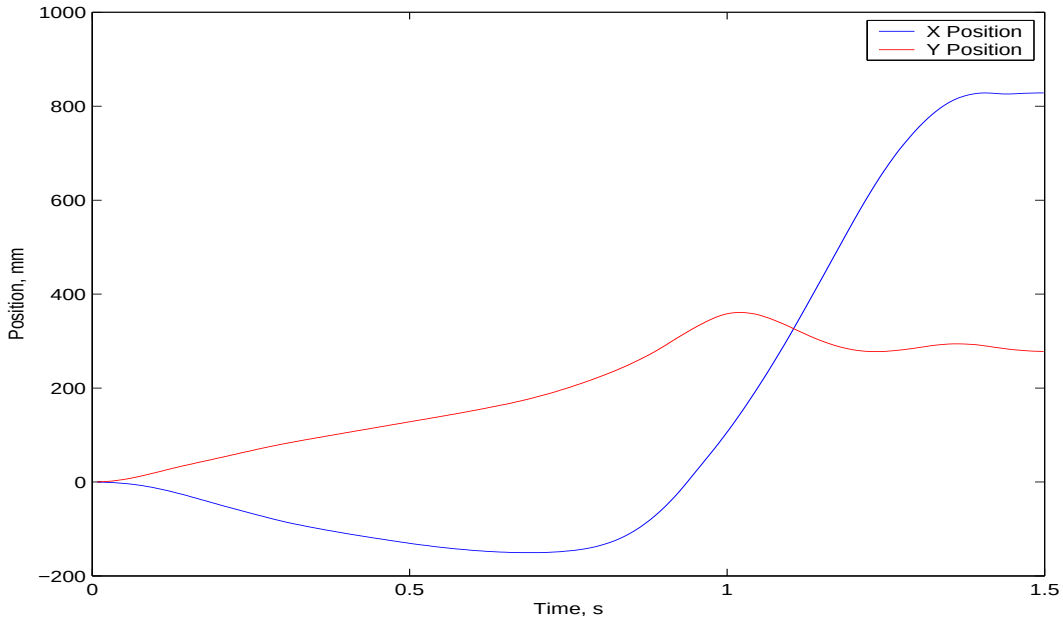


Figure 5.7: The resulting positions after integrating the acceleration curves using the algorithm in Section 5.3

where $S_{mc}$ is the motion capture step length and $S_{et}$ is the e-textile calculated step length.

The final two observations presented involve another combination of the step length algorithm with the motion capture and e-textile data. Figures 5.18-5.22 show the simulated data when the angle searching algorithm was used to calculate the initial angle of the subject's leg. The motion capture data was used for the initial velocity constants but not for the initial angle constant. This allowed the optimum initial angle to be found. For the e-textile data, this method provided the most accurate step length results of any other attempted thus far. Figures 5.23-5.27 show this data. The initial angle found was $-20.39°$ compared to the motion capture initial angle of $-27.90°$ that was used in the previous examples, shown in 5.23. When comparing Figure 5.12 with Figure 5.23, a few things can be noticed. The former has a consistent error from the motion capture angle throughout, while the latter has a larger error for the first half of a second, but then converges and follows the motion capture angle almost exactly for the last half of a second. The significance of this is that

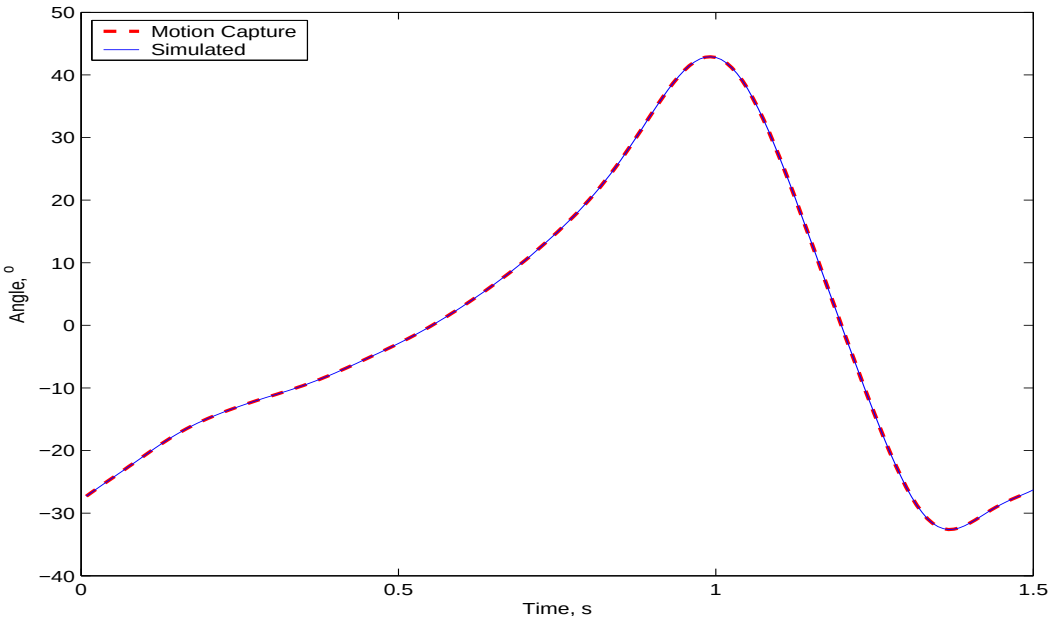


Figure 5.8: Simulated angle data compared with motion capture angle data with all integration constants known.

the angle change corrected the negative part of the acceleration that caused the large error in former step length. This resulted in a step length of $1167mm$ and an error of 12.7%, compared to an initial angle of $-30.17°$, a step length of $886mm$ and an error of 51% as shown in Figures 5.19 and 5.18 when this situation was simulated. Since the results for the simulated motion capture data are so erroneous, it can be concluded that this method is not very accurate for this step. Specifically, the angle searching algorithm did not produce correct results in simulation.

For the e-textile data, notice the spike in the X acceleration near the $1s$ mark in Figure 5.24. When the spike was integrated, it caused the X velocity in Figure 5.26 to deviate from the motion capture data. This spike corresponded to the point in time when the leg was at its maximum angle of orientation, as seen in Figure 5.23. This was the time directly after the heel leaves the ground. It was also the spike that the piece-wise linear fit developed in [4] was aimed at correcting. Furthermore, this spike was seen consistently throughout walking data

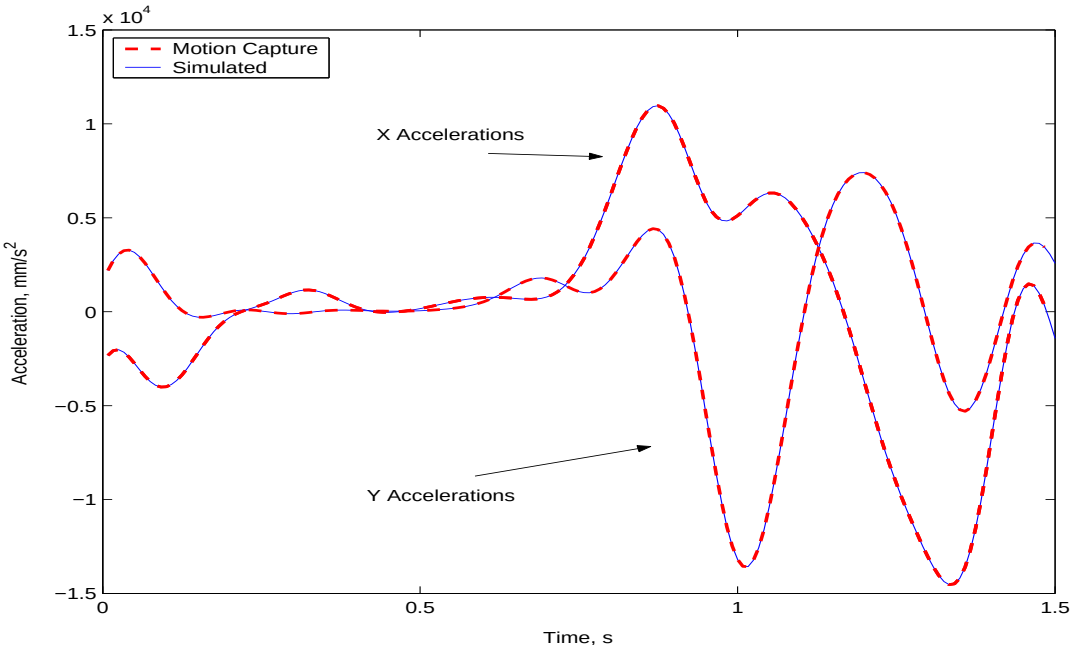


Figure 5.9: Simulated acceleration data compared with motion capture acceleration data with all integration constants known.

and can be seen at regular intervals in Figure 5.28. The spikes during the heel strike were caused by the shock of the sensor hitting the ground. Similarly, the smaller spikes during the heel lift off occur because of the sudden movement of the sensor.

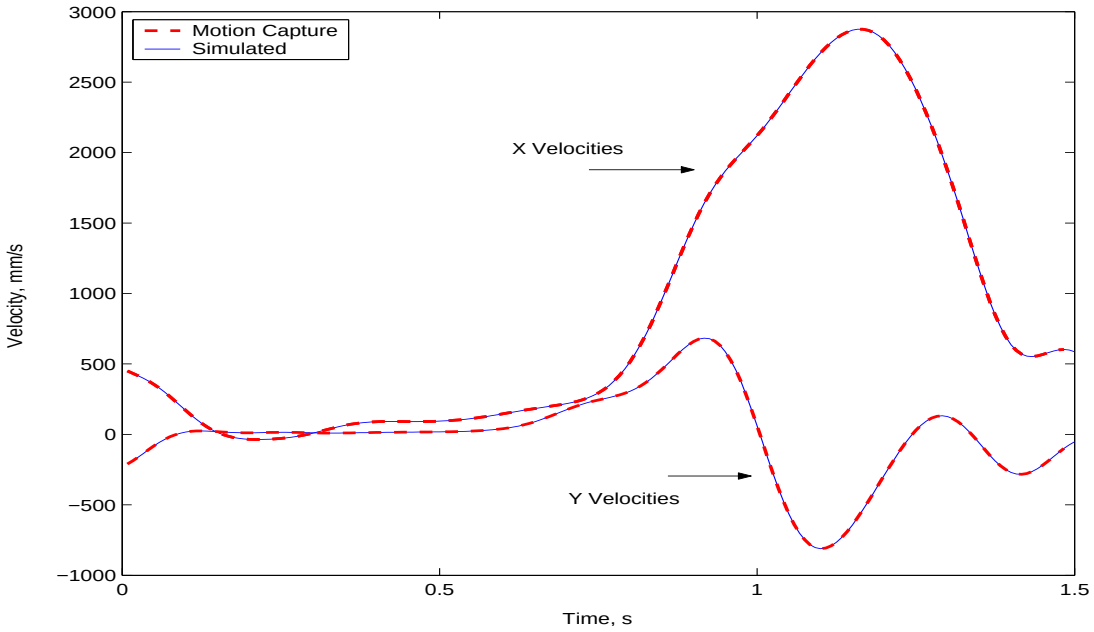


Figure 5.10: Simulated velocity data compared with motion capture velocity data with all integration constants known.

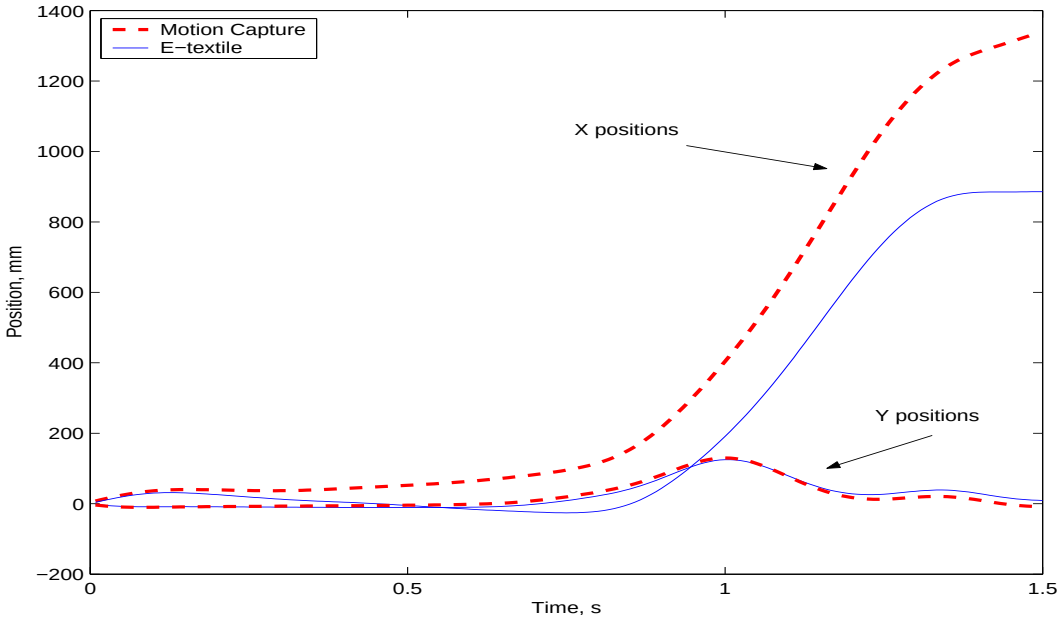


Figure 5.11: Simulated position data compared with motion capture position data with all integration constants known.

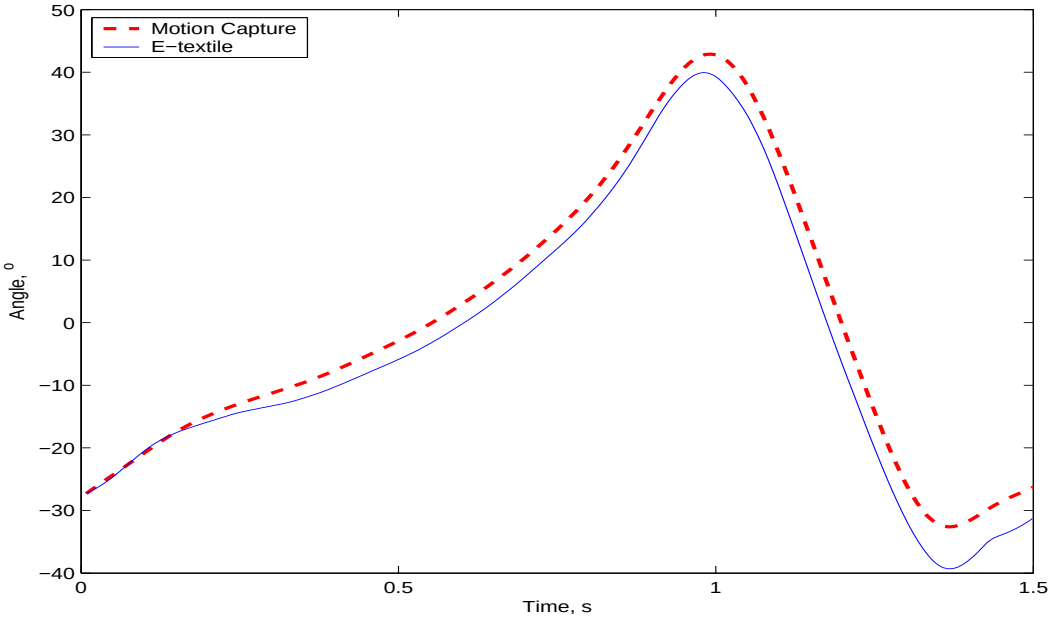


Figure 5.12: E-textile angle data compared with motion capture angle data with all integration constants known.

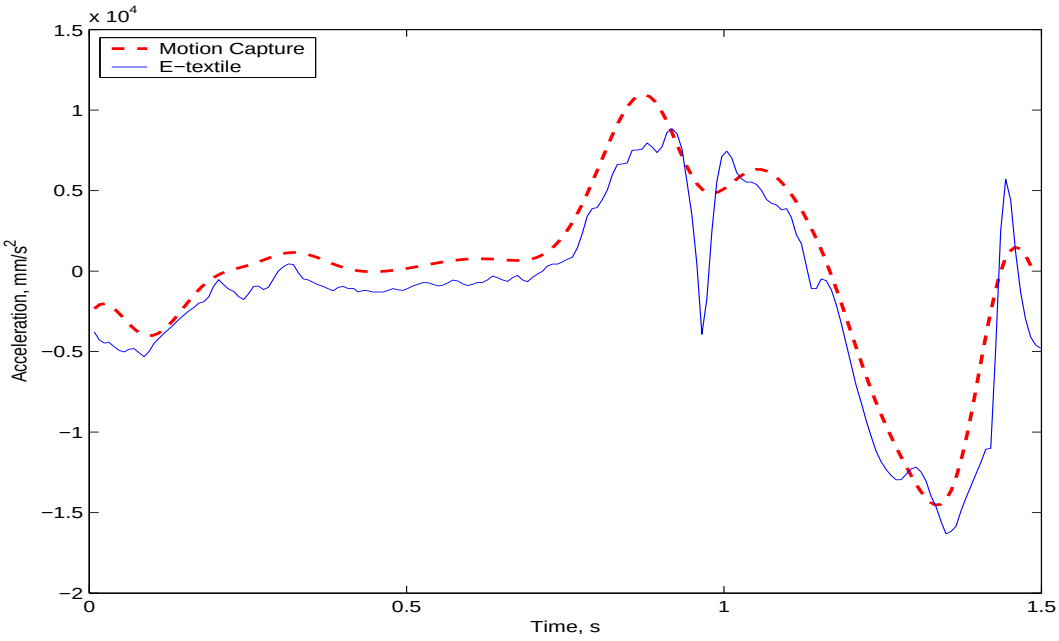


Figure 5.13: E-textile X acceleration data compared with motion capture X acceleration data with all integration constants known.

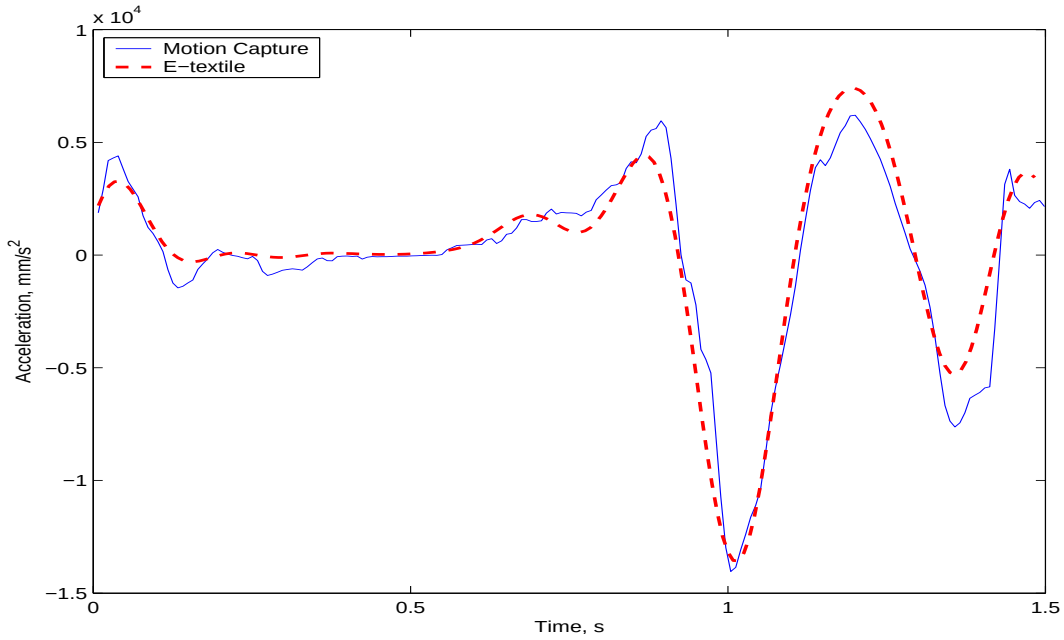


Figure 5.14: E-textile Y acceleration data compared with motion capture Y acceleration data with all integration constants known.

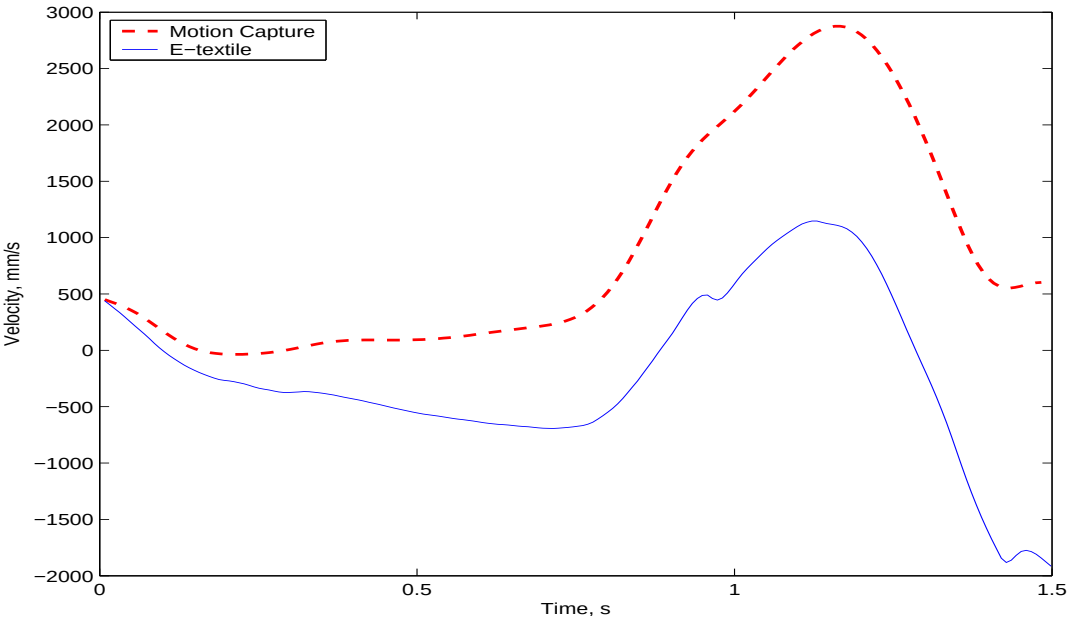


Figure 5.15: E-textile X velocity data compared with motion capture X velocity data with all integration constants known.

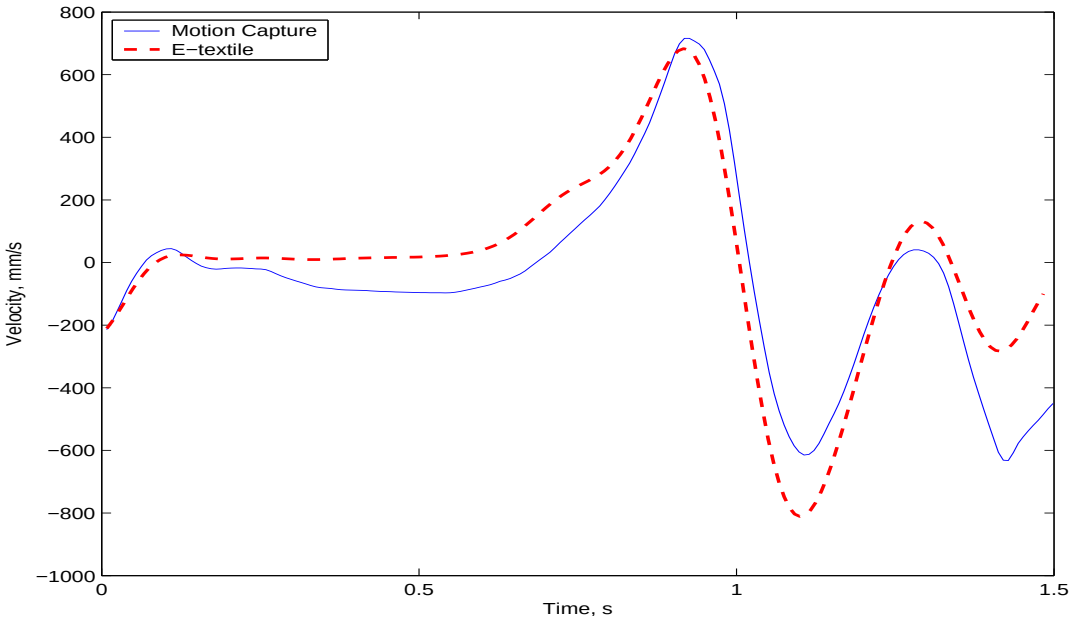


Figure 5.16: E-textile Y velocity data compared with motion capture data with all integration constants known.

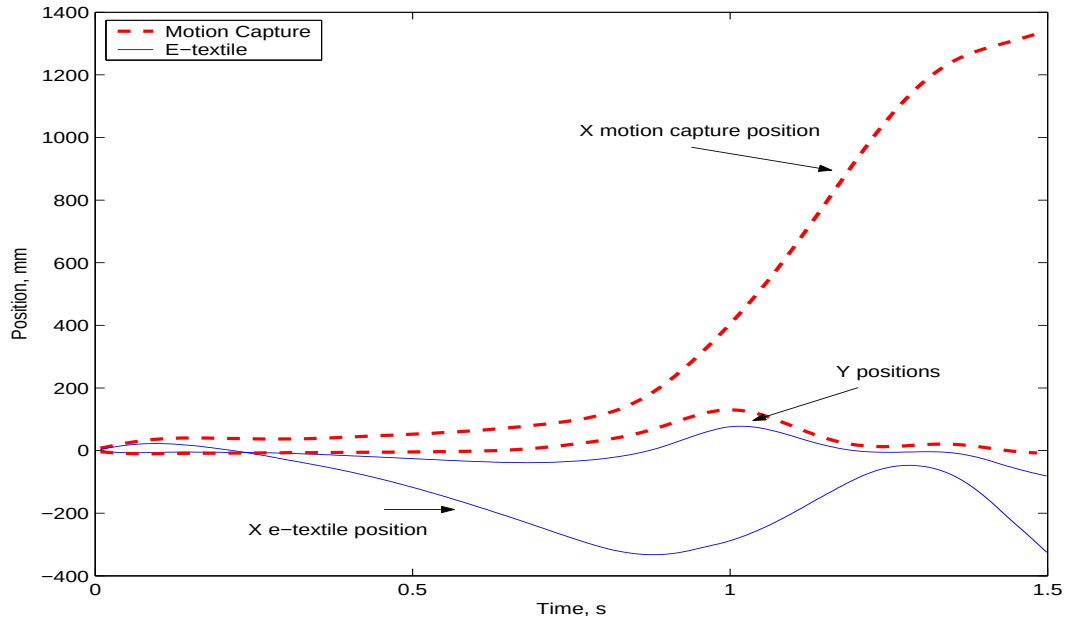


Figure 5.17: E-textile position data compared with motion capture position data with all integration constants known.

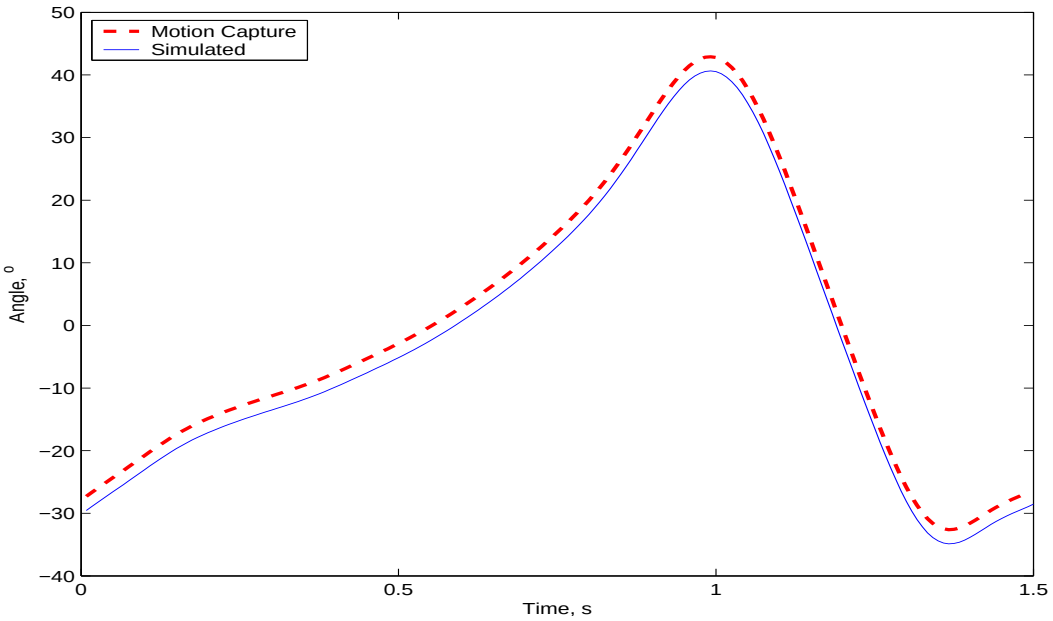


Figure 5.18: Resulting angle when the combination of motion capture initial velocity and angle searching algorithm used with simulated data

In designing the step length algorithm, a few assumptions were made. The most important of those was that at the beginning of a step, the initial values of the X and Y velocities were zero. From looking at the motion capture data and the above results, that is not a valid assumption. That assumption was also used when designing the searching algorithm to find the initial angle. Since the initial X velocity was not zero, Equation 5.1 had to be revisited. If $V_{x0}$ is the initial velocity then Equation 5.1 becomes:

$$\eta = |V_x - V_{x0}| + |P_y| \tag{5.10}$$

where $V_x$ is the ending velocity, $V_{xt}$ is the motion capture velocity at the end of the step and $P_y$ is the y position at the end of a step.

When this equation was used with the above e-textile data, the initial angle found became

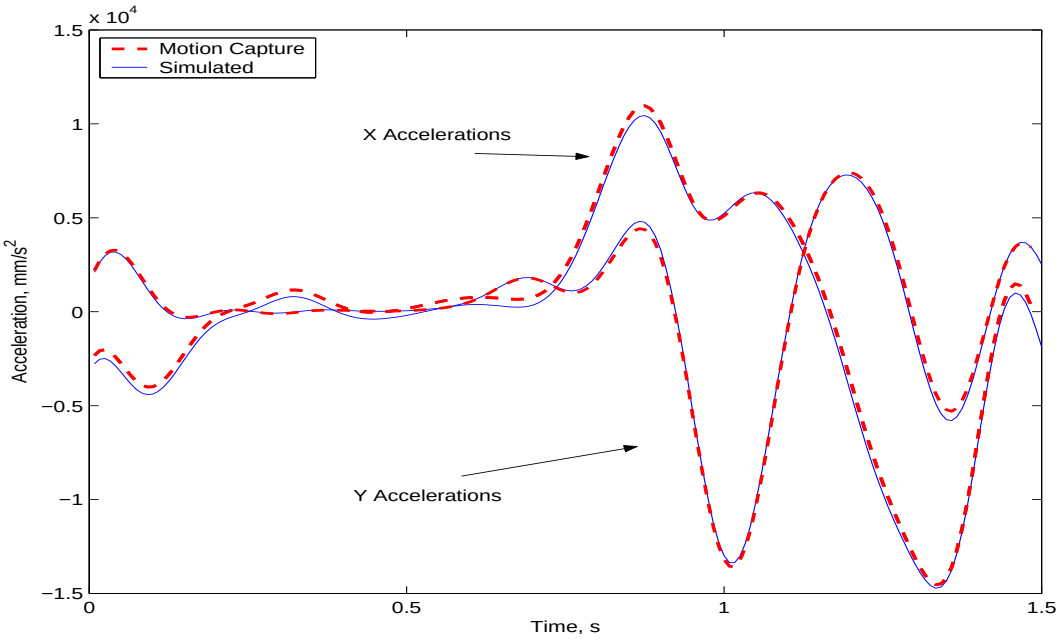


Figure 5.19: Resulting accelerations when the combination of motion capture initial velocity and angle searching algorithm used with simulated data

$-18.55°$. The corresponding step length was $1532mm$, and the error was 14.6%, actually increasing the error slightly. For the the simulated data, this change significantly improved the results. The initial angle found was $-28.36°$ with a step length of $1247mm$ and an error of 6.3%.

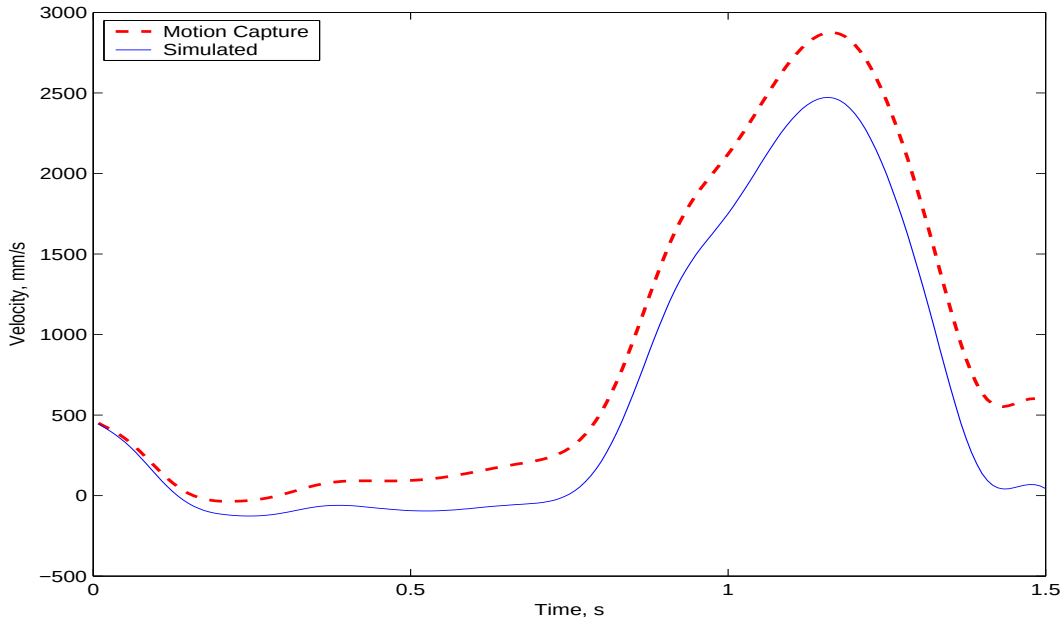


Figure 5.20: Resulting X velocity when the combination of motion capture initial velocity and angle searching algorithm used with simulated data

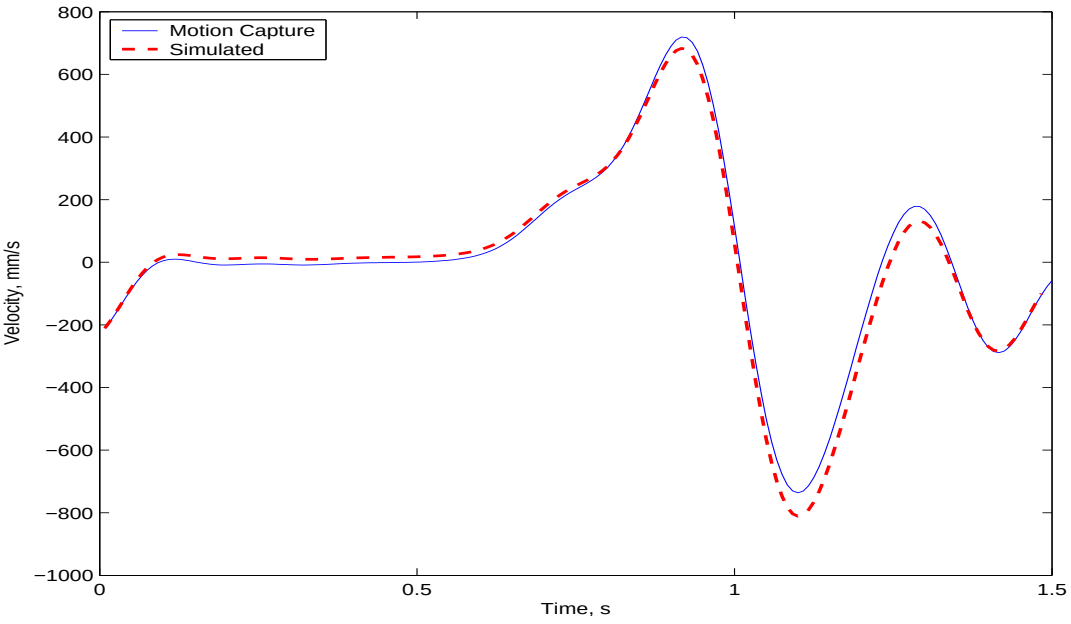


Figure 5.21: Resulting Y velocity when the combination of motion capture initial velocity and angle searching algorithm used with simulated data

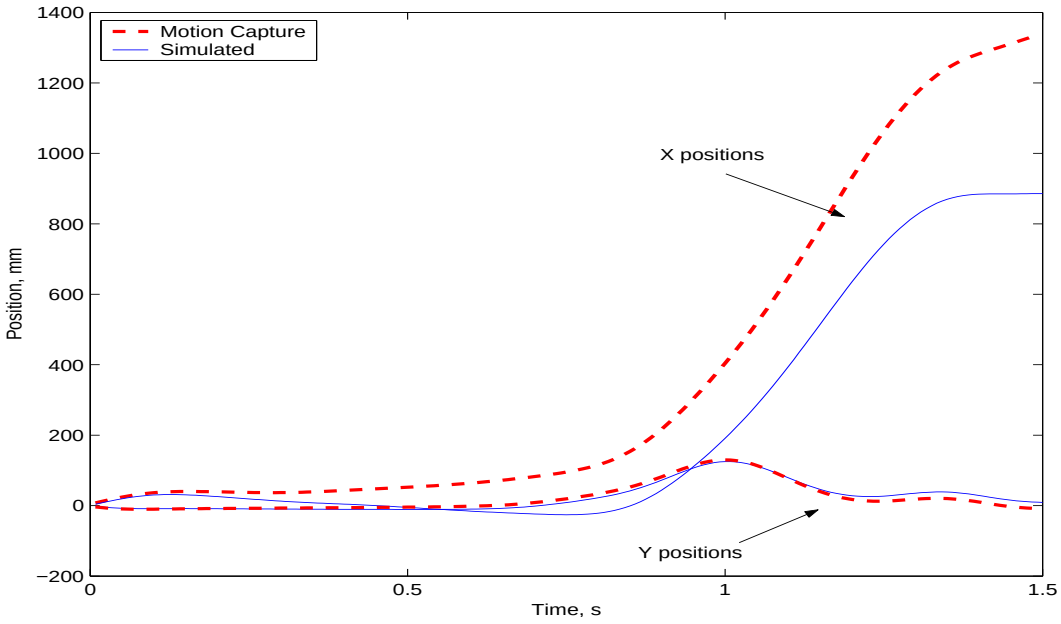


Figure 5.22: Resulting positions when the combination of motion capture initial velocity and angle searching algorithm used with simulated data

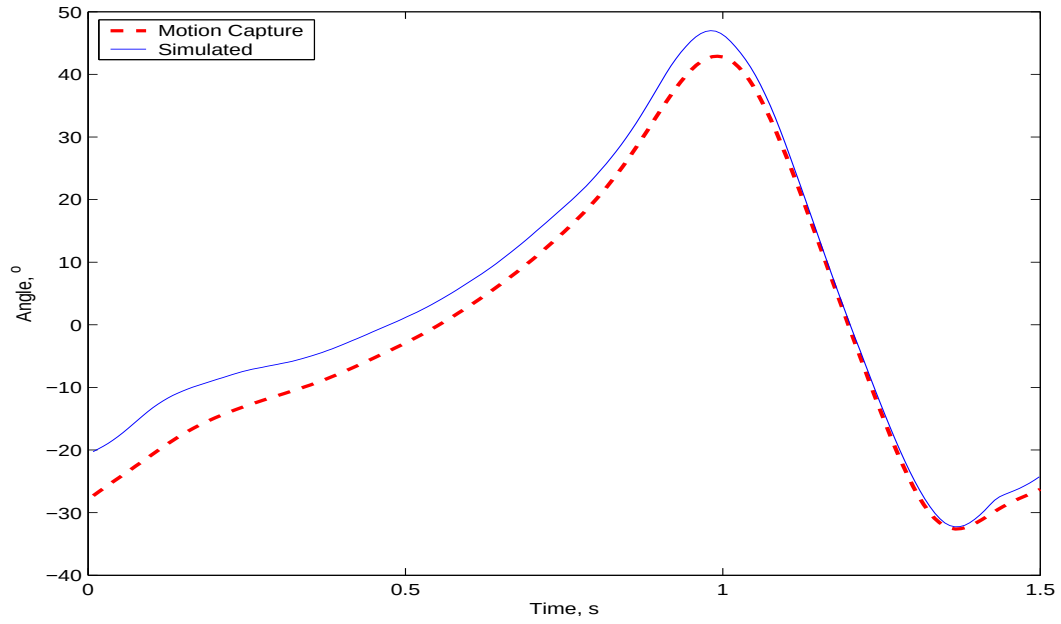


Figure 5.23: Resulting angle when the combination of motion capture initial velocity and angle searching algorithm used with e-textile data.

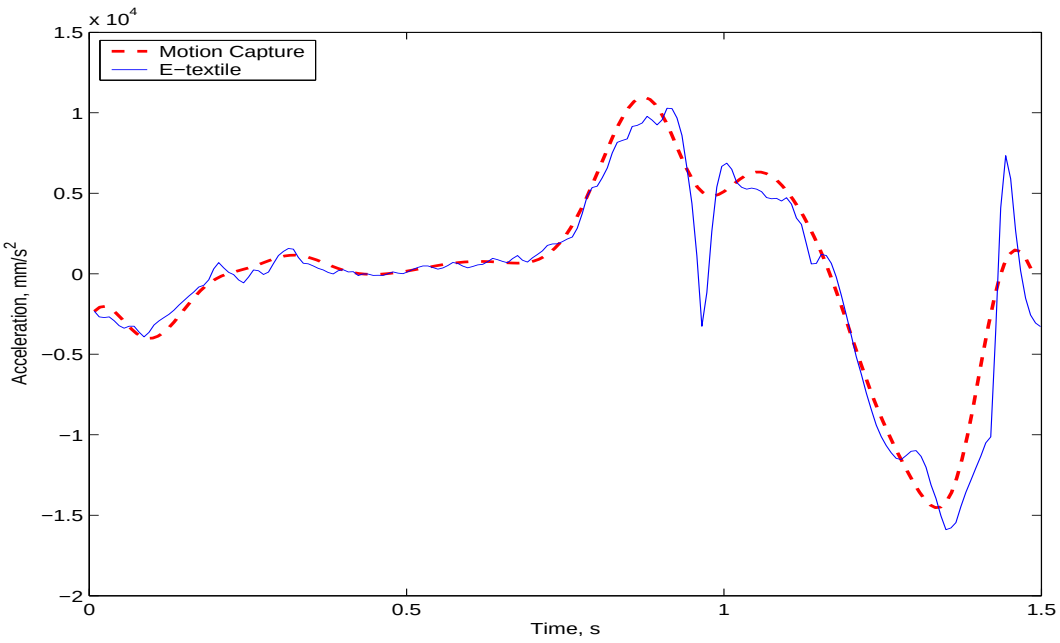


Figure 5.24: Resulting X acceleration when the combination of motion capture initial velocity and angle searching algorithm used with e-textile data.

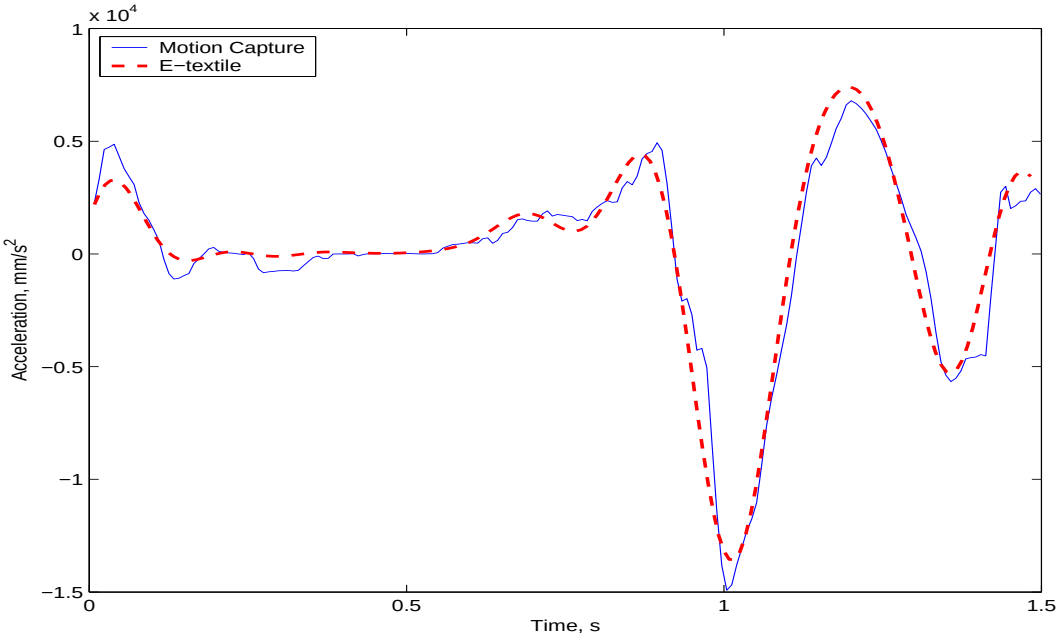


Figure 5.25: Resulting Y acceleration when the combination of motion capture initial velocity and angle searching algorithm used with e-textile data.

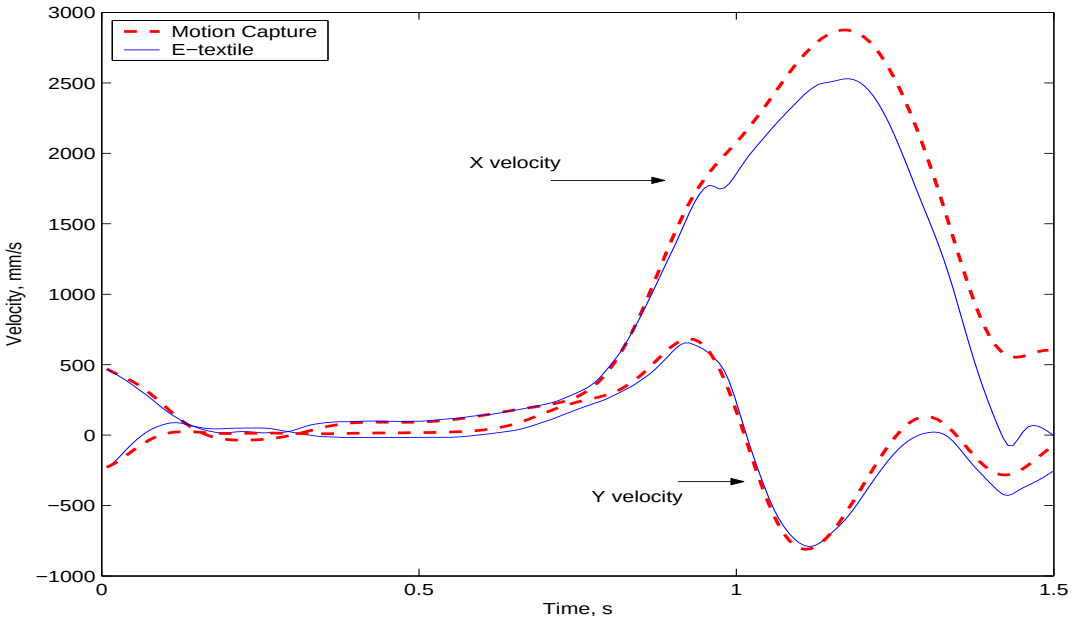


Figure 5.26: Resulting velocities when the combination of motion capture initial velocity and angle searching algorithm used with e-textile data.

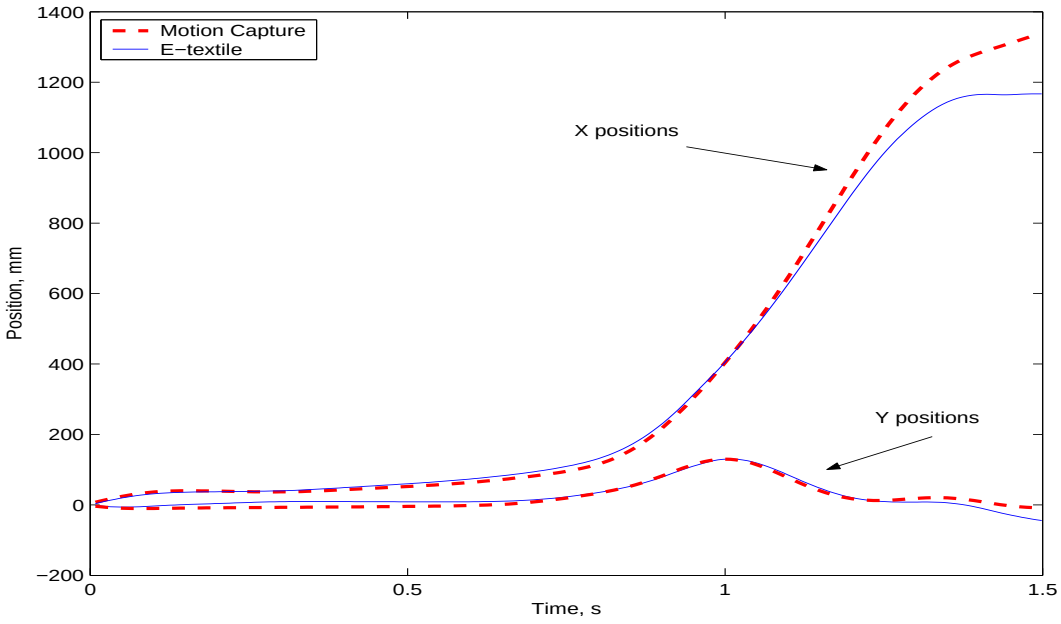


Figure 5.27: Resulting positions when the combination of motion capture initial velocity and angle searching algorithm used with e-textile data.

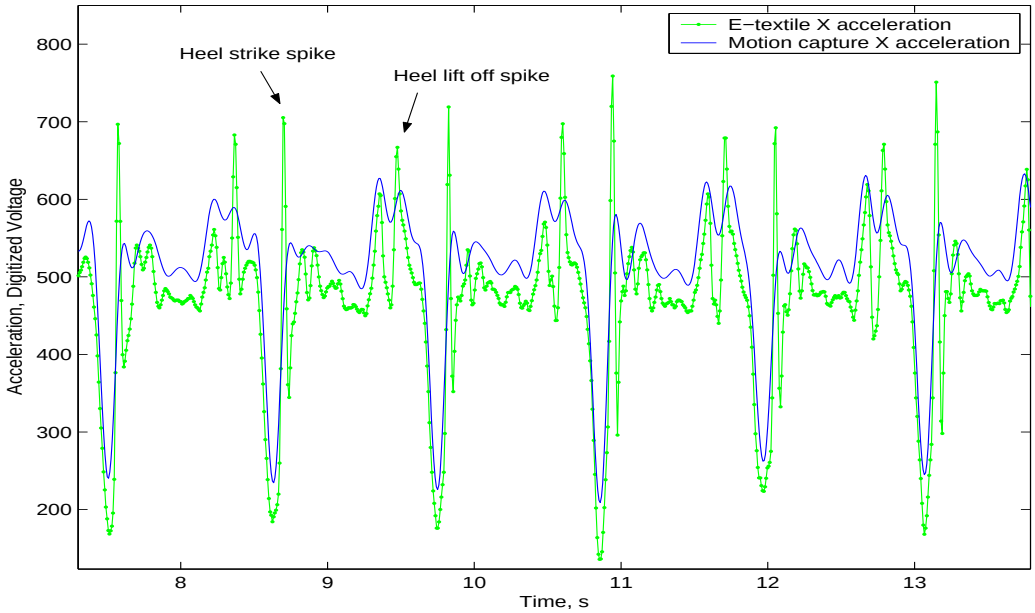


Figure 5.28: The X accelerations, showing the spikes the at both the heel strike and the heel lift off.

# Chapter 6

# Conclusion

This work described a simulation environment for e-textile motion sensing applications. The e-textile garment and its components were described in Chapter 3, along with the communication network incorporated with in the garment. Simulation models were generated for both a rate sensing gyroscope and a dual axis accelerometer. These models were compared directly to motion capture data in order to evaluate the error associated with the models. The sensor models for the ankles were accurate within 10%-20% of each other depending on the sensor and the axis. Reasons for this error were discussed in Section 4.5.3. These included errors in initial angle of the accelerometer, errors caused by the heel strike and errors caused by the sensors exceeding their specific dynamic range. All models were fundamentally dependent on the ability to approximate the angle of orientation of the point being modeled. Errors in this approximation, as seen with the knee data, directly caused inaccuracies in both the angular velocity and the acceleration approximations.

The designed simulation environment was then applied to attempt to calculate the gait parameter of step length. Errors in the approach to generate the step length parameter were discussed. In order to accurately calculate step length, three initial values used for integration

constants must be known: the initial velocity in the X direction, $V_{x0}$, the initial velocity in the Y direction, $V_{y0}$, and the initial angle of the leg, $\theta_0$. The original approach of assuming $V_{x0}$ and $V_{y0}$ were both negligible proved to be false. Simulated data was used to analyze the effects on the system of knowing more or less information about the integration constants. It was shown that the most accurate way to calculate step length was when the angle searching algorithm was adjusted to take into account $V_{x0}$. Fusing data from multiple sensors posed another problem in the step length calculation. Each sensor reading has a certain amount of noise and error associated with it. As the data from these sensors was combined, this error increasingly had a negative effect on the step length calculation. Also, two spikes in the accelerometer readings caused errors in the calculation, one that corresponded to the heel strike and another that corresponds to the heel lift off.

## 6.1 Future Work

There is much future work to be done with these applications. Based on the error of the sensor models, the models would be more accurate with the use of different sensors. The ones used here did not provide enough accuracy or dynamic range to sense all of the accelerations or angular velocities during walking. Three dimensional accelerometers and gyroscopes have enormous potential for accuracy when applied to both applications, because acceleration in all dimensions is fully constrained. Because of this, they can be calibrated without any user interaction or position requirements [30], and furthermore, any offset in initial orientation can be detected and corrected for, as long as the subject was stationary. The prototype e-textile garment still has room for great improvements. As of now, each e-TAG must be individually programmed and only performs one specific function. The current drive is towards a service based architecture that is fault tolerant and on the fly reprogrammable.

Though the results for the step length calculation were less then ideal, using e-textiles for motion sensing applications is extremely promising. In the field of gait analysis, future work includes deriving alternate measures to the gait analysis parameters of step length, heel strike velocity, velocity of center of mass and required coefficient of friction among others. The goal would be to avoid integrating values from sensors such as gyroscopes and accelerometers but rather use some feature of those readings directly as gait measures. For the step length calculation, future research involves attempting to remove the unknown integration constants from the equation, as well as compensating for the spikes during heel strike and heel lift off. Hopefully, as the technology improves, the spikes will be less of an issue. Another encouraging application of e-textiles is the use of singular value decomposition for activity recognition.

# Acknowledgements

There are many people who made this thesis possible with their assistance, support and guidance.

I would like to thank Dr. Thomas Martin for being my committee chair, advisor and friend over the past 14 months. Also, I'd like to thank Dr. Mark Jones for providing endless knowledge and assistance during this research. Additionally, I would like to thank them both for providing an excellent work environment, and for keeping their sense of humor intact throughout. I'd like to thank Dr. Peter Athanas for serving on my committee and providing any lab support needed. Also, I'd like to thank Dr. Thurmon Lockhart for use of his motion capture lab, as well as Jian Liu for his assistance.

I would like to also thank all of the students in the Configurable Computing and E-textiles lab for their help and support throughout. Specifically, Josh and Dave for their endless hours of discussion and help, especially when I first arrived in the lab. Also, I'd like to thank Ben Muzal for being in every project group I was in during my graduate studies and especially for his clutch performance during my final days. Also, thank you to Meghan, Vineet, Chris Z., Braden, and Scott for their help and entertainment while I was in the cubby hole.

I'd like to thank all of my friends who provided support and good times while I was in Blacksburg: Justin, Will, Julie, Amy, Draper, Landman, Billy, Jerry, Tony, Aaron and many

others. I'd also like to thank Emily for her love, support and for generally putting up with me.

Last and most importantly, I'd like to thank my parents and family for being there for me in every way imaginable throughout.

This material is based upon work supported by the National Science Foundation under Grant No. CCR-0219809 and CNS-0447741.

# Bibliography


[1] M. Weiser, "The computer for the twenty-first century," *Scientific American*, vol. 265, no. 3, pp. 94–100, September 1991.

[2] T. Martin, M. Jones, J. Edmison, and R. Shenoy, "Towards a design framework for wearable electronic textiles," in *Proceedings of the Seventh International Symposium on Wearable Computing.* ISWC 2003, October 2003, pp. 190–199.

[3] ——, "Modeling and simulating electronic textile applications," in *Proceedings of the ACM SIGPLAN/SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES '04)*, June 2004, pp. 10–19.

[4] J. Edmison, "An e-textile system for motion analysis," Master's thesis, Bradley Department of Electrical and Computing Engineering, Virginia Tech, 2003.

[5] C. Einsmann, M. Quirk, B. Muzal, B. Venkatramani, T. Martin, and M. Jones, "Modeling a wearable full-body motion capture system," in *Proceedings of the Ninth International Symposium on Wearable Computing.* ISWC 2005, 2005, pp. 144–51.

[6] R. Shenoy, "Design of e-textiles for acoustic applications," Master's thesis, Bradley Department of Electrical and Computing Engineering, Virginia Tech, 2003.

[7] D. Marculescu, R. Marculescu, N. Zamora, P. Stanley-Marbell, P. Khosla, S. Park, S. Jayaraman, S. Jung, C. Lauterbach, W. Weber, T. Kirstein, D. Cottet, J. Grzyb, G. Troester, M. Jones, T. Martin, and Z. Nakad, "Electronic textiles: A platform for pervasive computing," *Proceedings of the IEEE*, vol. 91, pp. 1995–2018, 2003.

[8] S. Park, C. Gopalsamy, R. Rajamanickam, and S. Jayaraman, "The wearable motherboard: An information infrastructure or sensate liner for medical applications," *Studies in Health Technology and Informatics, IOS Press*, vol. 62, pp. 252–258, 1999.

[9] R. Paradiso, G. Loriga, N. Taccini, M. Pacelli, and R. Orselli, "Wearable system for vital signs monitoring," in *International Workshop on New Generation of Wearable Systems for eHealth*, December 2003, pp. 161–168.

[10] E. Post, M. Orth, P. Russo, and N. Gershenfeld, "E-broidery design and fabrication of textile-based computing," *IBM Systems Journal*, vol. 39, no. 3 and 4, 2000.

[11] M. Satyanarayanan, "Pervasive computing: Vision and challenges," *IEEE Personal Communications*, pp. 10–17, Aug. 2001.

[12] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *IEEE Workshop on Mobile Computing Systems and Applications*, Santa Cruz, CA, US, 1994.

[13] T. Martin, Class notes - ECE 5564 Wearable and Ubiquitous Computing, Virginia Polytechnic and State University, Spring 2005.

[14] A. Golding and N. Lesh, "Indoor Navigation Using a Diverse Set of Cheap Wearable Sensors," in *Proceedings of the Third International Symposium on Wearable Computing*, October 1999, pp. 29–36.

[15] K. Van Laerhoven and O. Cakmakci, "What shall we teach our pants?" in *Proceedings of the Fourth International Symposium on Wearable Computing.* ISWC 2000, October 2000, pp. 77–83.

[16] H. W. Gellersen, A. Schmidt, and M. Beigl, "Multi-sensor context-awareness in mobile devices and smart artifacts," *Mobile Networks and Applications*, vol. 7, pp. 341–351, October 2002.

[17] J. Farringdon, A. Moore, N. Tilbury, J. J. Church, and P. Biemond, "Wearable sensor badge and sensor jacket for context awareness," in *Proceedings of the Third International Symposium on Wearable Computing.* ISWC 1999, October 1999, pp. 107–113.

[18] J. Edmison, M. Jones, Z. Nakad, and T. Martin, "Using piezoelectric materials for wearable electronic textiles," in *Proceedings of the Sixth International Symposium on Wearable Computing.* ISWC 2002, 2002, pp. 41–48.

[19] S. Kim and T. E. Lockhart, "Effects of age-related changes in hamstring activation rate and heel contact velocity on the biomechanics of slips and falls," in *The Proceeding of the XVI Annual International Society of Occupational Ergonomics and Safety (ISOES)(Session 5-2)*, 2002, pp. 1–5.

[20] F. J. Imms and O. G. Edholm, "Assessment of gait and mobility in the elderly," *Age and Ageing*, vol. 8, pp. 261–267, 1979.

[21] J. B. Dingwell and J. P. Cusumano, "Nonlinear time seris analysis of normal and pathological human walking," *Chaos*, vol. 10, pp. 848–863, 2001.

[22] J. Liu, T. Lockhart, T. Martin, M. Jones, C. Einsmann, and V. Jolly, "Dynamic Stability of Motion Impaired Elderly Assessed by E-Textile Pants," in *Submitted to IAE conference.*

[23] D. Lehn, C. Neely, K. Schoonover, T. Martin, and M. Jones, "e-TAGs: e-Textile Attached Gadgets," in *Proceedings of Communication Networks and Distributed Systems: Modeling and Simulation*, January 2004.

[24] Atmel Corporation, "AVR 8-Bit RISC Products," 2004. [Online]. Available: http://www.atmel.com/products/AVR/

[25] Analog Devices Inc., "ADXL203 Dual-Axis Accelerometer," 2004, http://www.analog.com/.

[26] ——, "ADXRS300 Angular Rate Sensor," 2004, http://www.analog.com/.

[27] Measurement Specialties Inc., "LDT-052K Piezoelectric Film Sensor," 2004, http://www.msiusa.com/.

[28] Koninklijke Philips Electronics N.V., "The $I^2C$ Bus Specification," 2000, http://www.semiconductors.philips.com/markets/mms/protocols/i2c/.

[29] Mathworks, "MATLAB Spline Toolbox," 2006, http://www.mathworks.com/products/splines/.

[30] P. Lukowicz, H. Junker, and G. Troster, "Automatic Calibration of Body Worn Acceleration Sensors," in *Proceedings of the Second International Pervasive Computing Conference*, 2004.

[31] H. Weinberg, "Using the ADXL202 Duty Cycle Output," Analog Devices Inc., Tech. Rep., 2002.

[32] CMU Graphics Lab Motion Capture Database, "http://mocap.cs.cmu.edu/search.html," 2003.

[33] J. M. Bland and D. Altman, "Statistical methods for assessing agreement between two methods of clinical measurement," *Lancet*, vol. 1, pp. 307–310, 1986.

[34] B. S. Everitt, *The Cambridge Dictionary of Statistics*. New York, NY: Cambridge University Press, 1998.

[35] W. J. Krzanowski and F. H. C. Marriot, *Multivariate Analysis, Part 1*. London: Edward Arnold, 1994.

[36] Maxim Integrated Products, "Converting between RMS and peak-to-peak jitter at specified BER," 2003. [Online]. Available: http://www.maxim-ic.com/appnotes.cfm/appnote_number/462

# Appendix A

This section shows the additional results for the left leg and the right knee data.

Table A.1: Correlation between e-textile and motion capture data for left ankle data

| | Pearson Correlation | 2-Tailed Significance |
|---|---|---|
| $A_x$ | 0.824 | 0 |
| $A_y$ | 0.694 | 0 |
| $\omega$ | 0.966 | 0 |

Table A.2: The limits of agreement and its confidence interval for the left ankle data

| | $d$ | | | $\lvert d \rvert$ | | |
|---|---|---|---|---|---|---|
| | avg | std | error | avg | std | error |
| $A_x$ | 61.419 | 72.227 | 0.380 | 74.280 | 58.920 | 0.310 |
| $A_y$ | 32.410 | 80.767 | 0.425 | 56.029 | 66.592 | 0.350 |
| $\omega$ | -11.052 | 44.334 | 0.233 | 33.714 | 30.838 | 0.162 |
| | Limits of Agreement | | 95% confidence interval | | | motion capture |
| | lower | upper | lower | upper | n | range |
| $A_x$ | -80.146 | 202.984 | -81.436 | 204.274 | 36126 | 887.383 |
| $A_y$ | -125.893 | 190.714 | -127.336 | 192.156 | 36126 | 648.112 |
| $\omega$ | -97.948 | 75.843 | -98.739 | 76.635 | 36126 | 730.125 |

Table A.3: The real and relative error of the two methods after removing the bias, with respect to the range of the motion capture data for the left ankle data.

| | std error | real error | relative error |
|---|---|---|---|
| $A_x$ | 0.658 | $\pm$145.112 | $\pm$0.164 |
| $A_y$ | 0.736 | $\pm$162.270 | $\pm$0.250 |
| $\omega$ | 0.404 | $\pm$89.073 | $\pm$0.122 |

Table A.4: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left ankle $\omega$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.539 | 1.599 | 6 | 26 | 0.187 |
| speed*age | 0.597 | 1.846 | 6 | 26 | 0.129 |
| speed*sex | 0.413 | 1.128 | 6 | 26 | 0.374 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.5: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left ankle $A_x$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.682 | 2.241 | 6 | 26 | 0.071 |
| speed*age | 0.378 | 1.010 | 6 | 26 | 0.440 |
| speed*sex | 0.569 | 1.723 | 6 | 26 | 0.155 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.6: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left ankle $A_y$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.425 | 1.170 | 6 | 26 | 0.353 |
| speed*age | 0.420 | 1.150 | 6 | 26 | 0.362 |
| speed*sex | 0.666 | 2.162 | 6 | 26 | 0.080 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.7: Tests of between subject effects of independent variables, left ankle $\omega$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0 | 1 | 0 | 0.097 | 0.765 |
| | maximum | 3.01e-05 | 1 | 3.01e-05 | 0.002 | 0.970 |
| | std dev | 3.44e-05 | 1 | 3.44e-05 | 0.143 | 0.717 |
| sex | mean | 0.002 | 1 | 0.002 | 1.547 | 0.254 |
| | maximum | 0.001 | 1 | 0.001 | 0.043 | 0.842 |
| | std dev | 0 | 1 | 0 | 0.860 | 0.384 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.007 | 7 | 0.001 | | |
| | maximum | 0.137 | 7 | 0.020 | | |
| | std dev | 0.002 | 7 | 0 | | |

Table A.8: Tests of between subject effects of independent variables, left ankle $A_x$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.001 | 1 | 0.001 | 0.494 | 0.505 |
| | maximum | 0.002 | 1 | 0.002 | 3.049 | 0.807 |
| | std dev | 0.000 | 1 | 0.000 | 2.085 | 0.437 |
| sex | mean | 0.006 | 1 | 0.006 | 2.3682.368 | 0.168 |
| | maximum | 0.002 | 1 | 0.002 | 0.076 | 0.791 |
| | std dev | 6.47e-05 | 1 | 6.47e-05 | 0.116 | 0.744 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.016 | 7 | 0.004 | | |
| | maximum | 0.195 | 7 | 0.007 | | |
| | std dev | 0.004 | 7 | 0.001 | | |

Table A.9: Tests of between subject effects of independent variables, left ankle $A_y$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0 | 1 | 0 | 0.058 | 0.816 |
| | maximum | 0.040 | 1 | 0.040 | 0.891 | 0.377 |
| | std dev | 0.001 | 1 | 0.001 | 0.498 | 0.503 |
| sex | mean | 0.011 | 1 | 0.011 | 4.467 | 0.072 |
| | maximum | 0.088 | 1 | 0.088 | 1.947 | 0.206 |
| | std dev | 0.003 | 1 | 0.003 | 2.901 | 0.132 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.018 | 7 | 0.003 | | |
| | maximum | 0.316 | 7 | 0.045 | | |
| | std dev | 0.008 | 7 | 0.001 | | |

Table A.10: Correlation between e-textile and motion capture data for right knee data

| | Pearson Correlation | 2-Tailed Significance |
|---|---|---|
| $A_x$ | -0.312 | 0 |
| $A_y$ | 0.091 | 0 |
| $\omega$ | 0.783 | 0 |

Table A.11: The limits of agreement and its confidence interval for the right knee data

| | $d$ | | | $\|d\|$ | | |
|---|---|---|---|---|---|---|
| | avg | std | error | avg | std | error |
| $A_x$ | -64.509 | 218.910 | 1.153 | 184.181 | 134.757 | 0.710 |
| $A_y$ | 2.819 | 112.208 | 0.591 | 81.240 | 77.450 | 0.408 |
| $\omega$ | 2.363 | 112.570 | 0.593 | 81.665 | 77.512 | 0.408 |
| | Limits of Agreement | | 95% confidence interval | | | motion capture |
| | lower | upper | lower | upper | n | range |
| $A_x$ | -493.572 | 364.555 | -497.486 | 368.468 | 36050 | 807.730 |
| $A_y$ | -217.108 | 222.747 | -219.114 | 224.753 | 36050 | 521.780 |
| $\omega$ | -218.274 | 222.999 | -220.286 | 225.011 | 36050 | 727.704 |

Table A.12: The real and relative error of the two methods after removing the bias, with respect to the range of the motion capture data for the right knee data.

| | std error | real error | relative error |
|---|---|---|---|
| $A_x$ | 1.997 | $\pm$439.817 | $\pm$0.545 |
| $A_y$ | 1.024 | $\pm$225.440 | $\pm$0.432 |
| $\omega$ | 1.027 | $\pm$226.166 | $\pm$0.311 |

Table A.13: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right knee $\omega$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.171 | 0.406 | 6 | 26 | 0.868 |
| speed*age | 0.493 | 1.418 | 6 | 26 | 0.245 |
| speed*sex | 0.500 | 1.445 | 6 | 26 | 0.236 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.14: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right knee $A_x$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.446 | 1.243 | 6 | 26 | 0.317 |
| speed*age | 0.474 | 1.345 | 6 | 26 | 0.274 |
| speed*sex | 0.494 | 1.422 | 6 | 26 | 0.244 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.15: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), right knee $A_y$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.180 | 0.429 | 6 | 26 | 0.853 |
| speed*age | 0.230 | 0.562 | 6 | 26 | 0.756 |
| speed*sex | 0.250 | 0.619 | 6 | 26 | 0.714 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.16: Tests of between subject effects of independent variables, right knee $\omega$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.063 | 1 | 0.063 | 53.099 | 0.000 |
| | maximum | 0.589 | 1 | 0.589 | 33.348 | 0.001 |
| | std dev | 0.410 | 1 | 0.410 | 34.456 | 0.001 |
| sex | mean | 0.074 | 1 | 0.074 | 62.552 | 0.000 |
| | maximum | 0.615 | 1 | 0.615 | 34.811 | 0.001 |
| | std dev | 0.049 | 1 | 0.049 | 41.356 | 0.000 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.008 | 7 | 0.001 | | |
| | maximum | 0.136 | 7 | 0.019 | | |
| | std dev | 0.003 | 7 | 0 | | |

Table A.17: Tests of between subject effects of independent variables, right knee $A_x$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.028 | 1 | 0.028 | 3.331 | 0.111 |
| | maximum | 0.096 | 1 | 0.096 | 1.439 | 0.269 |
| | std dev | 0.005 | 1 | 0.005 | 2.616 | 0.150 |
| sex | mean | 0.066 | 1 | 0.066 | 7.955 | 0.026 |
| | maximum | 0.210 | 1 | 0.210 | 3.139 | 0.120 |
| | std dev | 0.011 | 1 | 0.011 | 6.413 | 0.039 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.058 | 7 | 0.008 | | |
| | maximum | 0.469 | 7 | 0.067 | | |
| | std dev | 0.012 | 7 | 0.002 | | |

Table A.18: Tests of between subject effects of independent variables, right knee $A_y$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.009 | 1 | 0.009 | 2.472 | 0.160 |
| | maximum | 0.017 | 1 | 0.017 | 0.358 | 0.569 |
| | std dev | 0.002 | 1 | 0.002 | 0.977 | 0.356 |
| sex | mean | 0.002 | 1 | 0.002 | 0.683 | 0.436 |
| | maximum | 0.037 | 1 | 0.037 | 0.789 | 0.404 |
| | std dev | 9.72e-05 | 1 | 9.72e-05 | 0.062 | 0.810 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.024 | 7 | 0.003 | | |
| | maximum | 0.329 | 7 | 0.047 | | |
| | std dev | 0.011 | 7 | 0.002 | | |

Table A.19: Correlation between e-textile and motion capture data for left knee data

| | Pearson Correlation | 2-Tailed Significance |
|---|---|---|
| $A_x$ | -0.461 | 0 |
| $A_y$ | 0.181 | 0 |
| $\omega$ | 0.768 | 0 |

Table A.20: The limits of agreement and its confidence interval for the left knee data

| | $d$ | | | $\|d\|$ | | |
|---|---|---|---|---|---|---|
| | avg | std | error | avg | std | error |
| $A_x$ | 33.145 | 228.903 | 1.204 | 185.398 | 138.281 | 0.728 |
| $A_y$ | 15.950 | 91.247 | 0.480 | 70.143 | 60.500 | 0.318 |
| $\omega$ | -1.154 | 116.679 | 0.614 | 86.016 | 78.843 | 0.415 |
| | Limits of Agreement | | 95% confidence interval | | | motion capture |
| | lower | upper | lower | upper | n | range |
| $A_x$ | -415.504 | 481.795 | -419.592 | 485.883 | 36126 | 811.803 |
| $A_y$ | -162.894 | 194.794 | -164.523 | 196.423 | 36126 | 480.622 |
| $\omega$ | -229.845 | 227.536 | -231.928 | 229.620 | 36126 | 730.125 |

Table A.21: The real and relative error of the two methods after removing the bias, with respect to the range of the motion capture data for the left knee data.

| | std error | real error | relative error |
|---|---|---|---|
| $A_x$ | 2.086 | $\pm$459.892 | $\pm$0.567 |
| $A_y$ | 0.832 | $\pm$183.325 | $\pm$0.381 |
| $\omega$ | 1.063 | $\pm$234.421 | $\pm$0.321 |

Table A.22: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left knee $\omega$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.642 | 2.049 | 6 | 26 | 0.095 |
| speed*age | 0.427 | 1.178 | 6 | 26 | 0.348 |
| speed*sex | 0.406 | 1.104 | 6 | 26 | 0.387 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.23: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left knee $A_x$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.549 | 1.640 | 6 | 26 | 0.176 |
| speed*age | 0.510 | 1.485 | 6 | 26 | 0.222 |
| speed*sex | 0.363 | 0.961 | 6 | 26 | 0.470 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.24: Tests of Within-Subjects Effects of independent variables (Pillai's Trace), left knee $A_y$

| Within Subjects Effect | Value | F | Hypothesis df | Error df | Sig. |
|---|---|---|---|---|---|
| speed | 0.709 | 2.377 | 6 | 26 | 0.058 |
| speed*age | 0.505 | 1.464 | 6 | 26 | 0.229 |
| speed*sex | 0.609 | 1.895 | 6 | 26 | 0.120 |
| speed*age*sex | 0 | . | 0 | 0 | . |

Table A.25: Tests of between subject effects of independent variables, left knee $\omega$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.036 | 1 | 0.036 | 14.881 | 0.006 |
| | maximum | 0.177 | 1 | 0.177 | 8.583 | 0.022 |
| | std dev | 0.017 | 1 | 0.017 | 6.529 | 0.038 |
| sex | mean | 0.039 | 1 | 0.039 | 16.144 | 0.005 |
| | maximum | 0.183 | 1 | 0.183 | 8.912 | 0.020 |
| | std dev | 0.016 | 1 | 0.016 | 6.121 | 0.043 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.017 | 7 | 0.002 | | |
| | maximum | 0.144 | 7 | 0.021 | | |
| | std dev | 0.019 | 7 | 0.003 | | |

Table A.26: Tests of between subject effects of independent variables, left knee $A_x$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 1.21e-05 | 1 | 1.21e-05 | 0.002 | 0.963 |
| | maximum | 0.095 | 1 | 0.095 | 2.381 | 0.167 |
| | std dev | 0.001 | 1 | 0.001 | 1.465 | 0.265 |
| sex | mean | 0.004 | 1 | 0.004 | 0.752 | 0.415 |
| | maximum | 0.233 | 1 | 0.233 | 5.847 | 0.046 |
| | std dev | 0.005 | 1 | 0.005 | 5.477 | 0.052 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.037 | 7 | 0.005 | | |
| | maximum | 0.279 | 7 | 0.040 | | |
| | std dev | 0.006 | 7 | 0.001 | | |

Table A.27: Tests of between subject effects of independent variables, left knee $A_y$

| Source | Measure | Type III Sum of Squares | df | Mean Square | F | Significance |
|---|---|---|---|---|---|---|
| age | mean | 0.003 | 1 | 0.003 | 1.370 | 0.280 |
| | maximum | 0.014 | 1 | 0.014 | 0.288 | 0.608 |
| | std dev | 0.001 | 1 | 0.001 | 0.367 | 0.564 |
| sex | mean | 0.005 | 1 | 0.005 | 2.319 | 0.172 |
| | maximum | 0.021 | 1 | 0.021 | 0.442 | 0.528 |
| | std dev | 0.002 | 1 | 0.002 | 0.502 | 0.501 |
| age*sex | mean | 0 | 0 | . | . | . |
| | maximum | 0 | 0 | . | . | . |
| | std dev | 0 | 0 | . | . | . |
| Error | mean | 0.016 | 7 | 0.002 | | |
| | maximum | 0.339 | 7 | 0.048 | | |
| | std dev | 0.021 | 7 | 0.003 | | |